

开源时代

OPEN SOURCE TIMES

2009年2月刊 总第六期

开源业界:

英国政府欲在全国部署开源软件
微软起诉Tomtom 打响Linux侵权第一案

社区扫描:

Debian GNU/Linux 5.0版特性预览
发行版发布: Arch Linux 2009.02
Linux基金会: 已做好反击微软准备

技术新知:

Linux平台主流文件系统性能评测
Debian Etch发行版中Xen虚拟化应用实例
独辟蹊径网络安装系列之Redhat衍生版系列



从GNU项目诞生, FSF成立, 再到GNU通用公共许可证的面世, 自由软件运动发展的如火如荼, 转眼25年过去了, 行业发生了翻天覆地的变化, 就连创始人也开始慢慢老去。假如没有Richard Stallman, GPL授权、GNU项目, 还有自由软件基金会是否会倒下? 假如Stallman退休后, 它们将会如何发展?

2009 路在何方? GNU项目25周年

行业观察:

MySQL两高层离职对Sun影响有多远
圆桌论道: 经济危机与开源人才
红帽重拳出击虚拟化 欲重分市场版图

专家访谈:

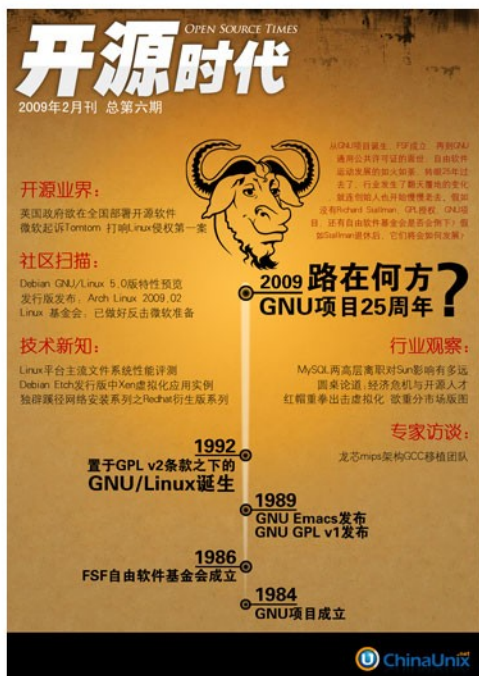
龙芯GCC团队

1992
置于GPL v2条款之下的
GNU/Linux诞生

1989
GNU Emacs发布
GNU GPL v1发布

1986
FSF自由软件基金会成立

1984
GNU项目成立



2009年2月刊 总第六期

编辑出品: 
网络发行: ChinaUnix

主编: 江晖
技术主编: 樊强
执行编辑: 周荣茂
内容编辑: 周荣茂 覃里
 李晓霞 周平
技术编委: 高延斌 马路遥
 白金
美术编辑: 林在子
交流论坛: bbs.chinaunix.net
杂志下载: www.chinaunix.net
 www.itpub.net
 www.ixpub.net

联系我们:
rmzhou@staff.chinaunix.net
投稿邮箱:
rmzhou@staff.chinaunix.net

媒体支持: 

广告联系: 温玉琴
电话: 010-82658790
手机: 13801339139
E-mail: wyq@it168.com

内容目录

卷首语

004

开源业界

MIPS科技加入Linux基金会	005
Novell第一季度净利润1070万美元同比跌36%	005
微软起诉Tomtom打响Linux侵权第一案	006
Android开源系统会让谷歌负担过重	006
安全机构建议美政府谨慎使用开源软件	007
戴尔: 三分之一的Mini9将运行Linux	008
中移动开放移动系统明年将投入商用	008
微软欲与Danger公司合作开源事宜	009
微软和Red Hat搁置争议 虚拟领域实现互操作	010
高通惠普12家公司加盟诺基亚Symbian协会	011
Marvell推出基于Linux系统的即插型计算设备	012
Zmanda发布开源备份云产品	012
Ubuntu欲用开源软件推动云计算	014
Symbian:Google Android开源只是幌子	015
英国政府欲在全国部署开源软件	015
微软务实对待开源	016

社区扫描

Debian GNU/Linux 5.0版特性预览	017
发行版发布: VectorLinux 6.0	018
Mozilla在SCALE上展示Firefox3.1的新特性	019
MySQL创始人正面回应离职原因	019
深入理解ubuntu 不仅仅再是桌面系统	020
Linux基金会: 已做好反击微软准备	021

Novell裁掉部分openSUSE Linux开发者	022
Ubuntu 9.10的开发代号被定为“Karmic Koala”	022
BASH 4.0发布	023
开源移动应用：谷歌苹果不相上下	023

行业观察

红帽重拳出击虚拟化 欲重分市场版图	024
俄罗斯是怎样发展Linux的？	028
Android与Symbian手机平台之争	028
MySQL两高层离职对Sun影响有多远	029
圆桌论道：经济危机、开源人才	030

专家专栏

龙芯GCC小组：积极参与社区建设	032
------------------	-----

本期推荐

GNU项目25周年，路在何方？	034
-----------------	-----

技术新知

独辟蹊径网络安装系列之Debian/Ubuntu	036
二探Cherokee：HTTPS主机及PHP支持设置	048
Linux操作系统下各种文件系统性能比拼	055
OpenBSD. Nginx. MySQL. PHP环境搭建手册	063
Linux系统下时间处理全过程	066
Ubuntu 8.10 中输入法ibus的安装和使用	068
Debian Etch发行版中Xen虚拟化应用实例	071

网友热评

086

版权声明

杂志内容来自ChinaUnix社区及互联网，电子杂志的宗旨是为了更好地传递开源最新自寻和技术经验。如有版权问题敬请联系，我们将会在第一时间做出处理。

致谢

本杂志得到ChinaUnix网站Linux时代社区版主的大力支持，技术文章大部分来自版主推荐，更多技术文章可以访问Linux时代精华区。本刊分析评论部分文章来自IT168技术频道。

卷首语

大家好，欢迎大家下载并且阅读 2009 年第二期《开源时代》杂志，总第 7 期。大家开始从假日中恢复过来，认真的投入到工作中去。今年的经济形势不容乐观，于是乎，大家都开始惊呼开源产业的春天即将到来。在他们看来，开源就是便宜货的代名词，是否如部分媒体和业界同仁所料，还有待这场危机和时间来验证。

且不论关系好坏，微软和开源两个词出现在一起的频率越来越高了。本月中旬，先是和红帽联合宣布 X86 平台虚拟化技术合作事宜，两周之后，专利侵权的大棒又举到了另外一家开源企业头上。微软的如此举动不得不令人咋舌。事实上，大家也都明白，微软作为一家企业，她的市场行为决定着他在业界的一举一动，更多的观点则是认为，这不过是微软的又一次 FUD 战术而已。

作为开源企业的领军人物，红帽也没有闲着，先是在其官方网站上打出整幅广告“用户的需求决定着合作”，暗示和微软的合作是来自客户的压力。是的，据说在 X86 服务器平台的操作系统市场，80% 的份额把持在微软和红帽两家手中。在如此的经济寒冬，虚拟化技术尘嚣日上之际，两者在虚拟化技术方面的合作，目的很明显，拒绝其他竞争对手入场。企业就是企业，利益才是最现实的目标。但是我们所庆幸的，这次合作，仅仅是技术支持方面的合作，并没有在软件版权等方面做出让步，这也许就是这次合作区别于上次微软-Novell 合作的根本所在吧。

要说到红帽的全新开源虚拟化解决方案，也是颇费一番周折，想当初，2006 年红帽推出 RHEL5 的时候，Xen 虚拟化技术可是红帽企业家操作系统的特色之一，随着 Xen source 和 Novell 的亲密，以及最终投向 Citrix 的怀抱，恐怕红帽现在想想还是有些后怕。对待后起之秀 KVM，红帽无论如何不能重蹈覆辙。08 年收购这家名为 Qumranet 的以色列公司，使得红帽真正的掌控了 Linux 平台虚拟化技术的发展话语权，毕竟 KVM 已经集成到了 Linux Kernel，这是其他 Linux 平台的虚拟化技术所无法做到的。

最后再来说说杂志稿件的情况，现在已经开始有不少杂志的读者给我们杂志投稿、提意见，我真的非常高兴。说明我们的杂志得到了用户的支持。同时，我们在稿件审查方面目前还存在着一定缺陷，但是网友能够指正我们的不足，我们也很高兴，杂志的成长壮大离不开大家的帮助，同样对我们的批评也是我们工作进步的重要因素！

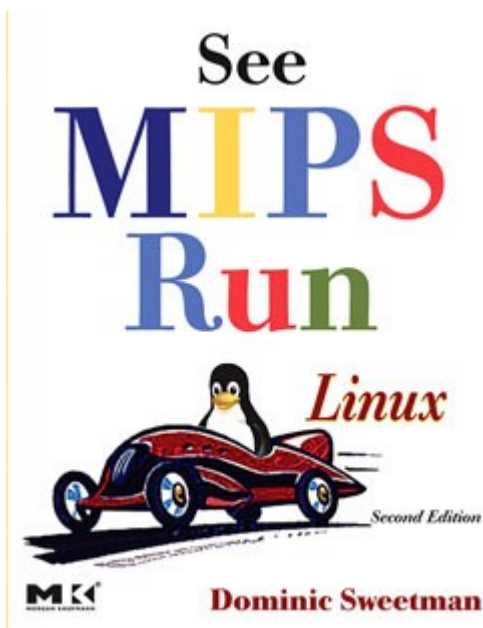
春暖花开，宅在办公室里的开源技术爱好者们、技术高手们，该出来走动走动了，身体是革命的本钱，开源事业的未来还需要大家的努力！

周荣茂

开源业界

MIPS 科技加入 Linux 基金会

致力于加速 Linux 技术成长的非营利组织 Linux 基金会 (Linux Foundation) 宣布, MIPS 科技公司 (MIPS Technologies) 已正式成为基金会的成员之一。MIPS 科技公司是为数字消费、家庭网络、无线通讯和商业应用提供业界标准架构、处理器及模拟 IP 的领导厂商。



加入 Linux 基金会之后, MIPS 科技将在其客户群及更广大的市场, 协助促进 Linux 的成长与推广。MIPS 科技加入的是一个由多家领先企业组成的机构, 这些企业都因客户充分发挥 Linux 与开放源技术的效益, 而获得业务成功与市场扩展。

MIPS 科技公司软件工程副总裁 UdiKalekin 表示: “大部分的 MIPS 开发人员都是采用 Linux 来进行产品开发。我们很高兴加入 Linux 基金会, 并由此更进一步持续并强化我们对 Linux 社群的贡献。我们预见, 未来支持多核心的 Linux 消费性平台将进一步演进, 而我们与 Linux 基金会的合作更将加速此一进展。”

Linux 基金会执行总监 JimZemlin 指出: “在如此艰难的经济情势下, MIPS 科技公司做出了一个睿智的决定。该公司对于 Linux 的投入, 以及愿意与其它业者、社群共同合作, 将有助于强化 MIPS 科技原本就已强大的市场地位。我们对 MIPS 科技公司未来对基金会的贡献, 深感期待。”

Novell 第一季度净利润 1070 万美元同比跌 36%

据国外媒体报道, Novell 北京时间 2 月 27 日, 即上周周四公布了 2009 财年第一季度财报。报告显示, 由于软件许可证营收减少 30%, Novell 第一季度净利润同比下滑 36%, 但每股盈利仍高于分析师预期。

报告显示, Novell 第一季度净利润 1070 万美元, 每股盈利 3 美分, 这一业绩不及去年同期。2008 财年第一季度, Novell 净利润 1680 万美元, 每股盈利 5 美分。

不计入一次性项目, Novell 第一季度每股盈利 7 美分, 这一业绩好于分析师预期。路透财经调查分析师此前平均预测, Novell 第一季度每股盈利 6 美分。Novell 第一季度总净营收 2.149 亿美元, 同比下滑 7%, 这一业绩不及分析师预期路透财经调查分析师此前平均预测, Novell 第一季度总净营收 2.3 亿美元。



Novell 维持此前目标, 仍预计 2009 财年调整后运营利润将增长 10% 以上。当日, Novell 在纳斯达克常规交易中报收于 3.30 美元。过去 52 周, Novell 最高价为 7.59 美元, 最低价为 2.49 美元。

微软起诉 Tomtom 打响 Linux 侵权第一案

北京时间 2 月 26 日消息，根据国外媒体报道，微软周三起诉 GPS 设备制造商 Tomtom 侵犯其 8 项专利权，该案被视为微软起诉 Linux 的第一案。微软周三称，已分别向华盛顿地方法院和国际贸易委员会(ITC)对 Tomtom 提起诉讼，指控后者侵犯其 8 项专利权。



Tomtom 的 GPS 设备采用的是 Linux 系统，尽管微软声称 Linux 侵犯其专利期已有多年，但该案被视为微软状告 Linux 侵权的第一案。

在微软所述的 8 项专利中，其中 5 项与车内导航技术有关，另外 3 项涉及文件管理技术。在这些专利技术上，微软已经与其他汽车导航设备厂商签署了授权协议。微软表示，仍希望与 Tomtom 在相关技术上达成授权协议。

早在 2007 年，微软就声称 Linux 及开源软件侵犯其 200 多项专利，此举导致微软与开源界关系紧张。当被问及“此举是否意味着微软将大规模启动 Linux 诉讼”时，微软当即给予了否认。微软称：“微软尊重也赏识开源软件所起到的积极作用，尊重开源开发人员对业界的贡献。”

Android 开源系统会让谷歌负担过重

美国知名科技博客 GigaOM 今天刊登分析文章称，虽然在很多人看来，开放与封闭之间的竞争毫无悬念——开放的平台总会获得最终的胜利。但是具体到移动领域，情况却并非如此。开放的谷歌 Android 与封闭的苹果 iPhone 之间的竞争存在着很多不确定性。Android 并不见得会像很多人所期望的那样轻而易举的取得这场竞争的最后胜利。

开放还是死亡

自从美国黑客、自由软件启蒙运动倡导者埃里克-雷蒙德(Eric Raymond)发表了那篇题为《大教堂与市集》(The Cathedral and the Bazaar)的文章后，“开放还是死亡”就成为了一种文化基因深深植根于人们的意识中。如果你不“开放”(例如，开源或开放 API)，就无法获得成功，而且将逐渐被人摒弃。尽管这一论述言之凿凿，但事实上，这并非是一个非黑即白的问题，涉及到移动领域时，更是如此。

谷歌和苹果就是一对很好的例子。尽管两家公司都掌握着旺盛的生态系统，而且在移动领域也都有很强的获利潜力，但是这两家有着巨大产品差异的公司对待开放的策略却迥然不同。苹果更加“封闭”，它将硬件、软件和服务进行了整合。而谷歌则较为开放，在对待系统和服务时，它采取的策略较为“松散”。这两家公司的移动产品都颇具突破性，而且也都吸引了大批追随者。那么究竟谁的方法更好呢？

iPhone VS Android

苹果的用户体验一向以来都为人津津乐道。它所提供的产品表现出优异的协同作用，而且不同的产品之间也实现了无缝结合。不过，苹果偶尔也会恃强凌弱，依据自己的喜好来选择为用户提供哪些附加价值，以及屏蔽哪些服务。

而谷歌所提供的产品门类则要广泛得多，不仅如此，它也更为开放，而且还借此获得了长足的发展。例如，谷歌允许第三方网站对谷歌地图服务进行优化，从而吸引到了大批第三方用户。然而，批评人士认为，谷歌的产品不具启发意义，而且从生命周期的观点来看，不够专注。

既然如此，就让我们把苹果 iPhone 与谷歌 Android 平台进行一下对比。

苹果 iPhone 为用户提供桌面、手机、网络和媒体等功能，而且将硬件、软件和服务进行了整合，不仅为用户提供了优异的体验，还借机发展了 iTunes 和“应用商店(App Store)”等新兴市场，并成功吸引了 2 万个应用程序以及 5 亿次的下载量。反对者则认为，苹果的方法太过封闭，因此注定要失败。

谷歌 Android 则采取了开源模式，它不仅没有与特定的硬件和服务提供商相整合，而且也不像苹果和 RIM 那样开发一个独立的手机品牌。

美中不足

力挺 Android 的人认为，Android 与 iPhone 的对决就好比当年微软与苹果在 PC 领域的战争一样。他们认为，苹果这样在硬件和软件方面都采取封闭策略的公司只能风光一时，而谷歌这样凭借一款软件与多家 OEM 厂商合作并“遍地开花”的企业才是真正的王者。在他们看来，这场手机大战的最终结局一定是以谷歌的胜利告终，届时，谷歌将把手机、PC 和网络完全整合起来。

然而，美中不足的，为了让设备制造商可以利用 Android 操作系统做“任何事情”，从而提供更好的用户体验，谷歌要么就要背负沉重的负担来支持“任何事情”，要么就要对部分功能进行限制。

当然，加以限制会降低 Android 的开放性，从而影响到整个生态系统的影响力。这是所有开源平台都会面临的问题，但是作为一款嵌入式的操作系统，Android 在移动领域所碰到的问题只会被放大。这是因为性能、可靠性和用户体验对于移动设备而言都至关重要，因此，“整合”就成为开源系统的关键所在。

胜负未定

事实上，开源只是一种属性，而并非结果，然而用户真正需要的是结果。用户希望获得整套的解决方案并且希望自己的工作具备足够的可预测性。只有很少一部分人会在乎产品的工作原理。iPhone 和 Android 这两大家族已经获得了消费者、开发人员和服务提供商的青睐，它们需要与硬件、软件和服务提供商进行更深层次的整合。

这并不是说这场竞争不会有结果，但是，我们却从中得到了启发，那就是谷歌未必是最终的赢家，而“开放与封闭”之间的对决也并不像某些人想象的那么毫无悬念。

安全机构建议美政府谨慎使用开源软件

根据国外最新报道，安全公司 Fortify 强烈希望奥巴马政府再三考虑，谨慎使用开源软件，尤其是涉及到安全方面的开源解决方案。

事件的起因，源于最近来自开源公司的 15 位高级主管游说奥巴马政府考虑采用开源解决方案。在这份游说人员名单中，既有来自 MuleSource、Compiere 公司的高管，也有来自 OpenLogic、Unisys 公司的高管。这些高管一致认为，“开源产业正在改变软件世界的方方面面，正如奥巴马承诺改变美国当前格局一样。我们真诚希望你们能使用开源软件，并把开源软件视为美国政府所倡议的进入新兴技术领域的一个关键组成部分。”

安全公司则表示，这其实不是一种好主意，因为在开源产品开发商里缺乏一种良好的安全行为模式。虽然安全公司没有明确谴责开源软件，但是这家安全公司的首席技术官 Roger Thornton 却警告说，通过未经授权的软件来节省成本其实会在未来导致更多的成本开支，比如维护成本、重新编



码成本和潜在的诉讼费用等。“如果安全目标不明确，或者安全措施不到位，那么，无论对开源软件或者商业软件来说，这种行为其实都是一种赌注。” Thornton 说道。这位曾经在苹果和 Sun 公司工作过的技术人士进一步表示，“我们在建立数百个组织并且在很多情况下，明确、制定工艺流程以确保应用安全性方面，都具有十分丰富的经验。考虑到开源软件的不安全因素，这些组织对安全控制已经十分到位。”

在西班牙，政府也正在鼓励采用开源软件，尤其是 Hipergate 软件——一种基于 Web 并可以运行在多个数据库和操作系统上的应用套装软件。Thornton 对此种解决方案深表忧虑，并希望美国政府避免类似的潜在风险。“无论是我们手工还是自动审查，都发现这个 Hipergate 缺乏足够的安全。这款软件每 1000 行代码中就有 16 个漏洞，这远远高出常规水平。”他说道。“正因为如此，我们强烈敦促奥巴马政府能彻底调查使用开源软件的可行性，并且也要考虑使用这种技术的所带来的影响。”

Fortify 是由一些行业专家组成一个专家顾问团，其中包括前微软人托德尼尔森(Tod Nielsen)和霍华德施密特(Howard Schmidt)，也包括在布什任内的前白宫网络安全顾问。巧合的是，微软也加入了这一场“安全”公关。

戴尔：三分之一的 Mini9 将运行 Linux

戴尔日前表示，其“上网机”Mini9 将有三分之一采用 Linux 操作系统。



上网机是眼下 PC 产业最亮丽的一道风景，也是 PC 业在全球经济衰退大背景下维系增长的主要产品。上网机通常可选择 Linux 或 Windows XP，不过有报道称，运行 Linux 的上网机的回报率要远远高于 XP 上网机，但戴尔并不赞同。笔记本电脑杂志的报道称，戴尔运行 Ubuntu（采用 Linux 内核）与 XP 操作系统的 Mini9 的回报率大致相当。戴尔的杰伊-皮克特表示：“戴尔三分之一的 Mini9 采用了 Linux，与戴尔其他运行 Linux 的产品相比，该产品的回报率略高，我们在用户宣传方面做了大量工作。”

戴尔将 Mini9 销售火爆的原因归结为 Linux 版本 Mini9 的低价格。

中移动开放移动系统明年将投入商用

据知情人士透露，中国移动开放移动系统(OMS)目前已开发完成，目前正在与相关硬件厂商进

行谈判与测试。“OMS 平台将于明年第二季度得到规模商用。”上述人士预测，目前正在 Marvell 平台上进行测试。

据中国移动官方信息，OMS 是基于 Linux 面向移动互联网的终端基础软件及系统解决方案，具有开放、安全、易用和友好用户体验等特性。

据传 OMS 的幕后开发者是播思公司，该公司于 12 月初加入开放手机联盟(OHA)。在金沙江创业投资基金的投资项目上，对播思公司简介中提及“播思与中国移动、Google、Marvell 成立了强大的多方技术联盟，共同开发和推广 OMS(开放移动系统)。播思的使命是实现 OMS 与中国移动的网络服务进行无缝对接的产品化。”对于这位合作伙伴的关系定位，中国移动相关人士则表示应以官方消息为准。

据了解，播思 CTO 饶宏曾是摩托罗拉 EzX 手机平台(基于 Linux)的创始人。12 月 19 日，在 CNGI 及移动互联网研讨会上，饶宏做了题为“图形用户界面框架与用户界面设计”的演讲，会场分析人士认为，饶的此次演讲可能为 OMS 后期针对客户的工作进行 UI 工程师的招聘或听取不同人士对 OMS 用户界面设计在做准备，此举意味 OMS 的开发已接近尾声。据见过 OMS 的人士透露，OMS 系统在易用及体验方面并不逊于其他操作系统。

会后，通信世界网就中国移动是否会推出基于 OMS 的 TD 定制手机一事向中国移动内部人士咨询，该人士私下表示 OMS 未来会应用到 TD 手机上，但中国移动并不会让 TD 手机独享 OMS，其他平台都适合 TD 手机，“况且这是终端厂商自己的事情，中国移动不会干预。”

微软欲与 Danger 公司合作开源事宜

微软正在寻找从事 Danger Sidekick 手机研发的 NetBSD 开发人员，这一动向或许表明微软会接受开源。微软可能会与 Danger 合作，通过招聘手机软件与移动业务开发人员，来完成开源。



业内有很多开发人员都对此议论纷纷，因为有很多人都说他们收到了招聘从事 Danger Sidekick 手机的 NetBSD 程序员的通告。听起来可能颇具戏剧性，Sidekick 用户可能都还没察觉到，程序员们也可能还没机会随心所欲地为这种设备创造新的应用程序。NetBSD 是一个开源操作系统，包括了许多的设备，例如，电脑和手机。但是，尽管它是开源的，它的许可证也不会强迫微软把自己的研发成果贡献出来。这是极具革命意义的一项举措，如果微软选择了一个继承了开源义务的平台，那这样的产品可能更能引起程序员的兴趣。

NetBSD 采用的是 BSD 授权许可证，在该许可证框架内，微软不会被要求分享其开发成果。然而，NetBSD 可以成为开源系统的一个根据，如果微软致力于某产品的开发，而这一产品在其核心技术里涉及 NetBSD，并且它允许程序员访问 NetBSD 应用程序接口。那么有可能微软会围绕 NetBSD 建立起全封闭的系统。

有分析家指出，微软的开源举措令人呢惊讶。微软走向开源的举措简直是对其商业模式进行的一个激进转变。微软一贯是通过出售软件来盈利。然而，它最近已经接受了一些开源的观念，例如，保持开放源码技术并且继续与开源社群分享开发。它可是长期与开源社群格格不入的。当微软收购

Danger 后，忠实的用户想知道原来手机的软件会不会转变为 Windows。许多人对可能发生这种转变报以遗憾的态度。他们抱怨 Windows 手机作为商务工具，其用户界面并不怎么好。Sidekick 是 Danger 最知名的产品，这类手机用户群一般是年轻人。转变到 NetBSD 不一定能减轻他们的忧虑反而可能会是一个难受的过程，因为微软很有可能把它驾轻就熟的 Windows 用户界面带入 NetBSD 的平台中。

但是 NetBSD 可以为微软提供工具上的帮助。因为 NetBSD 易嵌入，功能强大，容易维护，安全且以小轨迹运行。另外，对于微软来说它使用这些是免费的，它唯一要尽的义务是给予合适的信用。进一步的证据已经证实微软可能正在开发一个新的基于 NetBSD 操作系统，评论员们也指出一份写于 2007 年的文章中曾暗示 Danger 在微软收购其公司前就已经在把自己的软件转向 NetBSD。有可能，微软决定继续 Danger 在 NetBSD 上所做的工作。

Danger 公司的公共关系代表拒绝承认对 NetBSD 的使用。Danger 的操作系统和应用程序与后端服务器串联，用来提供诸如游戏，社交网络，互联网接入，Web 邮箱和即时信息的服务。摩托罗拉公司和夏普公司是 Danger 元件的生产商。

微软和 Red Hat 搁置争议 虚拟领域实现互操作

多年来，微软一直坚持开源软件厂商承认其专利是实现互操作的先决条件。微软本周一宣布在虚拟软件互操作方面与 Red Hat 进行合作。

据国外媒体报道称，双方之间的合作包括 3 个要点：Red Hat Enterprise 的虚拟技术将支持 Windows Server；Windows Server Hyper-V 和 Microsoft Hyper-V Server 虚拟技术将支持 Red Hat Enterprise Linux 服务器；完成测试工作后，双方将向同时使用双方技术的客户提供联合支持服务。微软虚拟战略部门总经理迈克·内尔称，客户需求是微软和 Red Hat 在虚拟技术互操作性方面合作的主要原因。

微软与 Red Hat 在虚拟技术互操作性方面的合作与专利问题没有任何牵连。根据协议，微软和 Red Hat 将向双方共同的客户提供联合技术支持服务，不要求双方共享知识产权。尽管之前也存在间接合作，但这是微软和 Red Hat 在互操作方面的首次直接合作。合作表明微软在如何与开源软件厂商打交道方面更成熟了，也表明 Red Hat 在如何与专有软件厂商打交道方面更成熟了，双方都意识到，合作才能实现双赢。



为满足客户的强烈需求，公司签署了与微软公司相互合作协议，此协议使两家公司虚拟化平台的互操作性得以增强。两家公司将加入彼此验证或认证项目，同时也将为他们共同的服务器虚拟化客户提供技术支持。此次相互验证可以使客户放心地部署异构的，虚拟红帽及微软应用方案。

红帽公司业务发展部副总裁 Mike Evans 表示：现今的 IT 世界是虚拟化和非虚拟化环境的混合体。红帽公司正在帮助我们顾客更迅速地延伸进入虚拟化环境，包括红帽企业 Linux 和 Windows 服务器的混合环境，红帽公司听到客户要求提供虚拟化解决方案的互操作性。所以今天我们很高兴宣布与微软的合作协议。

在此次公告发布后，红帽和微软公司的客户将可以在双方任何一个主机环境下运行微软 Windows 和红帽企业 Linux 虚拟服务器，并能得到这两个虚拟化和操作系统领导者的测试和支持。

这次相互合作协议的关键内容是：红帽将验证在红帽企业虚拟技术上 Windows 服务器虚拟机并获得技术支持。微软将验证在视窗服务器 Hyper-V 和微软 Hyper-V 服务器上的红帽企业 Linux 服务器虚拟机并获得技术支持。一旦公司完成了测试，客户在使用红帽企业平台上虚拟的 Windows 服务器操作系统时，或者使用视窗服务器 Hyper-V 和微软 Hyper-V 服务器虚拟的红帽企业 Linux 时，只要具备有效支持协议的客户将收到双方的合作技术支持。

微软虚拟化策略总经理 Mike Neil 表示：客户正迅速地采用 Windows Server 2008 Hyper-V 并且对能在虚拟化环境下在 Hyper-V 的操作系统上运行包括红帽在内更多的操作系统的做法非常有兴趣，我们欢迎红帽公司参与我们的服务器虚拟化验证项目并且期待通过与红帽的持续合作能让我们的顾客在其异构的 IT 环境下得到彼此技术支持的好处。

IDC 企业虚拟软件研究经理 GaryChen 认为：物理硬件并不关注在它上面安装了什么操作系统，而 hypervisor 提供的虚拟硬件更是没有大的差别。大部分顾客运作异构网络，因此这两个主要平台的互操作性会使企业客户更容易部署 IT 环境虚拟化。IDC 研究报告表示，Windows 和红帽企业 Linux 是两个主要的企业部署的操作环境，占运行在 hypervisors 上 x86 操作系统的 80%，故此，我们欣喜地看到这两大平台的供应商能把他们的竞争搁置在一边而把顾客放在首位。

此协议为微软和红帽共同客户使用服务器虚拟化建立了协调的技术支持。在该协议中并不要求共享知识产权。因此，这项协议除工业标准的验证/认证测试费用外，并不包括任何专利或开源许可证权利及额外财务条款。在此互认证和支持协议下的红帽和微软服务器虚拟化解决方案的验证，将在今天开始执行，预计将在今年下半年发布第一批结果。

高通惠普 12 家公司加盟诺基亚 Symbian 协会

据国外媒体报道，高通、惠普、MySpace 及其它 12 家公司近日加盟诺基亚主导的 Symbian 协会(SymbianFoundation)，在手机软件战中，该协会规模超过了谷歌的 Android 阵营。

在过去两年中，由于谷歌和苹果进军手机市场，该市场开始向手机软件研发转型。诺基亚的 Symbian 操作系统占据着手机软件市场的领先地位，但其市场份额在过去一年中遭受苹果、RIM 和其它后来者蚕食。诺基亚为应对上述竞争者，于去年收购了智能手机软件商 Symbian 剩余的所有股权，并承诺免费向其它手机厂商转让手机软件。在此以后，共有 78 家公司表示计划加盟 Symbian 协会。去年 12 月，谷歌 Android 阵营公布了新成员，其成员总数达到了 47 个。

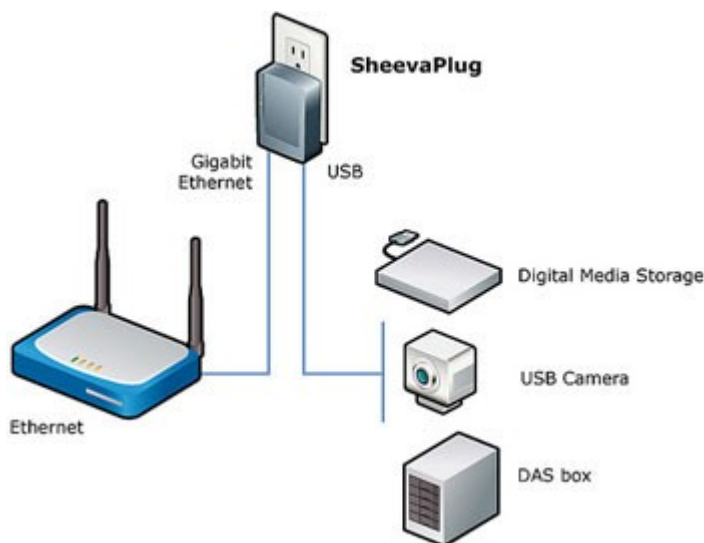
周四宣布将加盟 Symbian 协会的公司还包括 Atelier、美国银行、Gemalto、Imagination Technologies、Mobica、Nanoradio、欧姆龙、SanDisk、SESCA、SiRFTechnology 和 VirtualLogix。

诺基亚今后将 Symbian 的资产投入非营利的 Symbian 协会，并联合领先的手机厂商、网络运营商和通信芯片厂商创建一个开源平台。诺基亚已表示，把 Symbian 协会视为向市场带来新产品的更便捷途径，该协会成员也不用向协会外的软件研发方支付费用。诺基亚计划今年发布首款

Symbian 协会旗下的软件，并将在 2010 年 6 月前推出新的手机软件平台。

Marvell 推出基于 Linux 系统的即插型计算设备

近日，宣布推出 Plug Computing 计划，帮助开发人员和终端用户轻松获得高性能、环保且可始终保持开机和连接状态的计算体验。Plug Computer 身形小巧，能够直接插入壁式插座，且功耗极低，可始终保持开机。与其他家用嵌入式设备不同，Plug Computer 采用千兆赫级处理器，性能可与 PC 相媲美。这使得它能够替代 PC 实现任何软件应用。



根据介绍 SheevaPlug 公版设计包括有核心频率为 1.2GHz 的 ARM 处理器、512MB DDR2 内存、512MB flash 内存、千兆级网卡和一个 USB2.0 接口，能耗仅为 5W。至于软件方面，Marvell SheevaPlug 将可以支持“多种标准的 Linux2.6 内核”。价格方面也是非常诱人，售价也只有 99 美元，预订购买只要 79 美元。

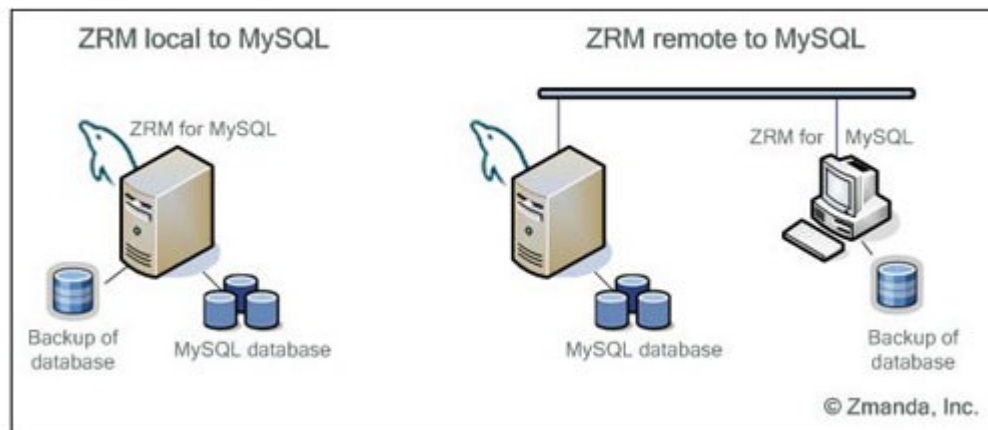
目前系统开发商可以通过 Marvell 公司的网站在 99 美元的价位上购得整套 SheevaPlug 开发包。至于其他的用户可能会更愿意等待一段时间，因为 Marvell 公司很快就会推出其他的配件，比如价值 79 美元的云计算引擎插件和 CTERA CloudPlug（云直连存储）。

Zmanda 发布开源备份云产品

Zmanda 本周推出了两款新产品，一款是联机备份与灾难恢复产品 Zmanda Cloud Backup，它适用于基于亚马逊 Simple Storage Service(S3)服务的微软 Exchange 和 SQL Server；另一款是 Zmanda 的开源 Amanda 计划开发出的最新版数据保护软件 Amanda Enterprise Version 3.0。

Zmanda 的首席执行官 Chander Kant 表示，Zmanda 备份的数据量仍在继续增长，第四季度比第三季度增加了 50%。他说：“我们的备份服务价格便宜，灵活性高，深受 IT 经理们的欢迎。”

Taneja Group 的分析师 Eric Burgener 表示，Zmanda 拥有一套完整的企业备份产品，就象赛门铁克的 NetBackup 或者 EMC 的 NetWorker 一样，虽然它的功能只有后者的 80% 左右，但是价格却只有后者的 30%。



它不支持各种高级功能如裸机整体恢复、SAN 无服务器备份、单备份目录同时对接多备份主机以及磁带备份，但是 Burgener 指出，小型企业通常不需要用到所有的高级功能。尤其是在当前这种困难的经济时期，Zmanda 的价值论相对于为某些客户量身打造的版权商品更有吸引力。

Burgener 说，Zmanda 的可调整性有限，因此它们并未打算向大型企业推销它们的产品。只有一个主备份服务器与各备份目录对接，也就是说它只能支持几百个系统，而不能支持上千个系统。而市场上的企业级备份产品如 NetBackup、NetWorker 或 IBM 的 Tivoli Storage Manager 可以在一个备份目录周围通过一个集中化管理点去管理多个备份主机。

Burgener 表示，Zmanda 的计划系统是其突出技术特色，在整个业界也相当独特。用户只要给出两个参数，比如希望多久对某个文件进行一次备份，计划系统就可以确定如何在资源需求有限的情况下进行备份，而且还可以根据环境的变化进行动态调整。他说："尤其是对偏重于产品易用性的中小型企业来说，这是一个巨大的优势。"

Burgener 指出，Zmanda 还有一个优势就是它使用了开放的协议和格式，这样就减少了支持和维护方面的问题，让数据恢复变得更加容易。

Enterprise Strategy Group 的分析师 Lauren Whitehouse 表示，Zmanda 对 UNIX 和 Linux 系统的支持非常突出，但是她补充说，它还需要增加对各种流行管理程序如 VMware、微软 Hyper-V 和 Citrix Xen 等的支持。

Whitehouse 说："Zmanda 的产品或许与常规不同，它备份的方式也许与通常做法稍有不同，比如将整个备份过程延伸到一周，通过这种方式来优化性能，而不象常规做法那样一周备份一次。"

她说："我认为 Zmanda 的目标是改变现状，即备份解决方案市场完全由版权格式和版权云产品垄断的现状。Zmanda 为市场提供了一种价格更低廉的解决方案，为企业用户提供了更多的选择余地。"

Amanda Enterprise 3.0 中增加了许多新功能，包括支持微软 SharePoint Server 2007、PostgreSQL 和 EnterpriseDB 的 Postgres Plus 系列产品的实时备份代理等。这款产品还支持磁盘到磁盘到磁带(D2D2T)以及磁盘到磁盘到云(D2D2C)备份，而且对于 Sun 用户来说，还增加了对 ZFS 文件系统的支持。对于 Windows 用户来说，增加了对远程备份和恢复的加密支持，Zmanda Management Console 可以自动查找磁带库以及对单个磁带库中的多个磁带驱动器进行管理的功能。

Amanda Enterprise 3.0 目前还处于测试阶段，很快就会通过 Zmanda 的网络商店公开发售，起步价为每位客户 100 美元和每个应用代理 300 美元。

Ubuntu 欲用开源软件推动云计算

Ubuntu 创办人 Mark Shuttleworth 近日宣布，Ubuntu 9.10 将以 Karmic Koala 为代号。一如往常，Ubuntu Linux 将分成两部分：桌面版和服务器版。桌面版的主要目标是“第一印象”，Shuttleworth 暗示“开机过程将令人惊艳”。他也承诺 Ubuntu 的外观设计将有显著改变。他说：“桌面版将充满设计师的苦心积虑，我们已开始认真地进行新外观的制作。Brown 让我们很满意，但 Koala 会考虑其它选择。”

我相信其它人也会以同样认真的态度，检视桌面版的设计。但 Shuttleworth 口中的服务器版功能，则让云计算社群更感兴趣。看来 Karmic Koala 服务器版的主要工作，将与云计算相关。以下是 Shuttleworth 这部分声明的完整内容：

“一只聪明的无尾熊（Koala）即使身在云雾中，也看得到整片树林。Ubuntu 要利用 Amazon EC2 的 APIs，将自由（Free）软件置于云计算的前锋，让任何人都能使用完全开放的工具，轻松设立他们自己的云端。我们目前正在测试 Ubuntu AMI 在 Amazon EC2 上执行。在 Karmic 期间，我们要简化应用软件在云端的布署，利用即可用的设备，或快速收集的订制影像。Ubuntu-vmbuilder 可用来轻松制作订制的 AMI，但标准影像档案集，可让 EC2 上从事类似工作的人，更容易协调合作。让 Amazon 丛林变得更容易通过，对 Ubuntu 而言不是更恰当吗？

如果你要自行建构一个 EC2 形式云端该如何？无尾熊最爱的是尤加利叶。UCSB 的 Eucalyptus（尤加利树）计划，能让你在自己的硬件上、为自己的资料中心制作一个 EC-2 形式云端。Eucalyptus 已加入整体的大计划，在 Karmic 期间，我们将让这些云端飞舞，根据各位的需求，动态地增减资源配置。聪明的无尾熊知道，保存精力最好的方式就是睡觉，现在就连服务器都可以中止和重启，你可以想象，我们能建构一个在午睡时间耗能降到近乎零，有工作再醒来的云计算设施。”

如果你看不懂，重点如下：

- 1、Ubuntu 服务器版的目标，是以完全开源的软件推动云计算；
- 2、针对那些有意管理云端的人，Ubuntu 将包含利用 Amazon APIs 优势的工具；（我希望 GoGrid APIs 也在考虑之列，因为这是多个 Amazon 竞争者采用的接口。）
- 3、Canonical 将根据 Karmic Koala 制作标准的 Amazon Machine Images，基本上是为 Amazon 社群制作相当于“标准构造”的“即可用”装备；
- 4、不想专用 Amazon？你宁愿用自己的基础设施建构一个云端，而舍弃好用的公共云端？从 Karmic 的前身 Jaunty Jaguar Jackalope（程序代码即将冻结）开始，UC Santa Barbara 的开源计划 Eucalyptus，将纳入每一个版本。

我确定 Canonical 现任的软件服务经理 Simon Wardley，对这个方向有很大的影响。他对开源标准在云计算领域的长期投入，让这些工具的选择显得十分合理。未来几个月内，Koala 包含的平台极可能改变，但 Shuttleworth、Wardley 和 Ubuntu 社区信奉的开放软件哲学，将主导他们的选择。这对云计算的未来有多重要仍待观察，目前已有其它开源计划含有“内建的”Amazon 影像（不只是 RedHat），而 Eucalyptus 是一个研究计划，需要进一步制作，才能发挥生产用途。

在我眼中，Ubuntu 团队就像是 Linux 的苹果，极可能改变整个局势，但不是透过基础建设本身，而是他们以建构和管理云端所创造的使用者经验。我迫不及待。

Symbian:Google Android 开源只是幌子

最近口水战真多，Symbian 和 Google 近日也因开源的问题起了争执，Symbian 首先指责 Google 根本是在打着开源的幌子推销 Android，Google 紧接着回应称 Symbian 才是虚伪的开源组织。

Symbian 组织主管 Lee William 表示，Android 只是打着‘开源’的幌子来掩盖其闭源的本质。Android 并不是开源，这只是种营销手段，它完全在 Google 的控制之下。

William 补充说：“这确实是一个很好的伪装，但是我不认为基于 Linux 就等同于开源。”根据 William 的说法，开源平台背后需要的是一个社区而不是一家公司，“如果你问我 Symbian 未来两年的蓝图，我只能说我不知道，它的方向是由社区所有成员共同决定的。”



Google 移动平台部门副总裁兼 Android 联合创始人 Rich Miner 则对 Symbian 开源性也进行了质疑，他说：“如果一个平台的源代码并不能完全向所有人公开，而只是面向某个组织的成员开放，那么我认为称这个平台是开源的是一种错误，因为其消费者比如手机原始设备制造商和电信网络供应商等不能对该平台进行修改、调整，如果你需要加入某个组织才能得到这个平台的源代码，那么这个平台并不是开源的。”

Miner 解释 Google 在 Android 平台所扮演的角色时表示：“Android 里并没有任何我们未公开的 Google 技术，我们的竞争对手也能获得和我们自己所知一样多的资源。”他也承认 Google 毕竟是一个企业，它和非营利性组织 Symbian 不同，所以对于 Google 来说“所有的东西都必须开放和免费”是不必要的。

英国政府欲在全国部署开源软件

根据 BBC 文章报道，英国政府周一宣布，将考虑在全国范围内开展开源软件推动措施，主要领域为公众服务，政府将考虑使用开源软件替代原有的商业软件。

英国政府表示，如果该项目的财政预算通过审核的话，就会在全国各个领域主要是公众服务方面大量采用开源软件。虽然开源软件大都是免费的，但是其后续开发、维护和支持却需要资金投入，这正是英国政府目前正在深入研究的。

关于当前使用的版权软件，英国政府头疼的是其合同和许可授权问题，一旦许可期满，政府就要再次支付大量资金，而开源软件则不存在这个问题。BBC 在报道中指出，采用开源软件除了能为政府节省 6 亿英镑的开支，还能获益于开源软件的稳定性和安全性。

关于全面部署开源软件的计划，英国政府并没有透露任何细节。这一决策自然会影响到很多软件供应商的利益，虽然包括 Sun 微系统的许多公司对此表示支持，不过最大的软件供应商微软目前持反对态度，不希望将自己的软件过渡为 Linux 之类的开源系统，公司表示希望英国政府能放弃这一计划。

微软务实对待开源

据国外媒体报道，在周三的斯坦福 Accel 研讨会上，不少支持开源的企业家，例如 SUN 公司的 Zack Urlocker，Cloudera 的 Mike Olson 都在大力推广开源思想。在他们之中，最出风头的，不是别人，正是微软负责服务器与开发工具事业部的高级副总裁 Bob Muglia。



Bob Muglia，微软负责服务器与开发工具事业部的高级副总裁 Alfresco 公司的首席技术官 John Newton 对此十分关注，并对 Bob 在大会上的陈述发表了自己的看法：

在某种程度上，几乎我们所有的产品将采取开源形式。如果 MySQL 和 Linux 操作系统做足够好，用户将理所当然地使用他们。事实上，目前很多的微软产品已经开始推行开源政策，比如 MSN 或是 VISUAL STUDIO，但是对于绝大多数的商用软件和专有软件，开源的推进还是不够深入。然而，开源是软件发展的主流这一点却是不可争议的。

2006 年，Gartner 曾谈及代码重用的重要性时，称将帮助开源的进一步推广。微软也表示，借用一些开放源代码要比凡事亲历亲为更加务实。由此可见，随着软件业的发展，“开源”已不再是微软眼中的毒瘤，而成为了一种有效快捷的开发方式。让我们满心欢喜的迎接微软回归到我们的现实中来。

社区扫描

Debian GNU/Linux 5.0 版特性预览

在经历了 22 个月坚持不懈地开发之后，Debian GNU/Linux 5.0 版(代号 Lenny)正式发布了。Debian GNU/Linux 是一个自由的操作系统，它支持 12 种处理器架构并带有 KDE、Gnome、Xfce 和 LXDE 桌面环境。同时它和 FHS v2.3 兼容，其软件针对 LSB 3.2 版开发。

Debian GNU/Linux 可以在多种计算机上运行，从掌上机和手持系统到超级计算机，在它们之间的几乎任何机型都可以。它支持十二种架构：Sun SPARC (sparc)、HP Alpha (alpha)、Motorola/IBM PowerPC (powerpc)、Intel IA-32 (i386)、IA-64 (ia64)、HP PA-RISC (hppa)、MIPS (mips, mipsel)、ARM (arm, armel)、IBM S/390 (s390) 以及 AMD64 和 INTEL EM64T (amd64)。



Debian GNU/Linux 5.0 Lenny 加入了对 Marvell 的 Orion 平台的支持，它被用于许多存储设备。所支持的存储设备包括 QNAP Turbo Station、HP mv2120 和 Buffalo Kurobox Pro。另外，Lenny 现在支持多种网络笔记本电脑 (Netbook)，特别是华硕出品的 Eee PC。Lenny 还包含了用于 Emdebian 的编译工具，用它可以对 Debian 源代码包进行交叉编译并收缩，使之适用于嵌入式 ARM 系统。

Debian GNU/Linux 5.0 Lenny 包括了新的 ARM EABI 移植，armel。这个新的移植可以更有效地使用现代和未来的 ARM 处理器。因此，老的 ARM 移植 (arm) 就过时了。

新引入的软件包可以完全地支持 NTFS 文件系统，并且能够直接使用绝大多数多媒体按键。通过 swfdec 或 Gnash 插件可以支持 Adobe Flash 格式的文件。对笔记本电脑的支持得到了全面提升，比如 CPU 频率自动调节的原生支持。新加入的几个游戏可用来消磨闲暇时光，包括解谜游戏以及第一人称

射击游戏。还有值得一提的是新增加的 goplay，它是一个图形化游戏管理器，提供了过滤器、搜索、抓屏以及对 Debian 中的游戏进行介绍等功能。

下面，我们就来介绍一下 Debian 5.0 的 5 个最佳功能：

1、集成 X.org 7.3

由于集成 X.Org 7.3，X server 可以对绝大多数硬件进行自动配置。过去，在 Linux 环境下配置屏幕确实是一件非常痛苦的事情。有了 X.org 7.3，程序就能自动设定显示分辨率。

2、安全方面进行了改进

Debian 的开发人员一直努力改善软件的安全性。Debian 5.0 在安全性方面的改进包括在安装后第一次启动之前就安全所有安全更新、减少标准安装可执行文件和打开的端口，以及使用 GCC 加强特性编译个对安全有严格要求的软件包。其他多种软件也都有特别的改进，比如 PHP 现在已经使用 Suhosin 的加强补丁编译。

3、Java 友好

Debian GNU/Linux 5.0 新加入了 Sun OpenJDK，GNU Java 编译器，GNU Java 字节码解释器，Classpath---一个开源的 Java 类库集合。这意味着你可以在 Debian 上使用大多数基于 Java 的程序。这些编译器和类库有些已经被预先加入了 Debian 的类库中，你立刻就能使用。

4、Debian 使得开源播放器和阅读器能够播放一些专有的媒体格式

利用 Moonshine 和 Moonlight 这样的程序，你可以非常容易得就能在 Linux 上播放私有多媒体影片。Debian 通过 swfdec 或 Gnash 插件可以支持 Adobe Flash 格式的文件，这是一个非常受欢迎的功能。

5、Debian 可以运行在任何平台上

如果你是开发人员，那么你当然可以让 Linux 操作系统运行在任何系统上，比如 iPhone 和微软 XBox 360。但是，如果你不是程序员，那么如果你想在 Sun SPARC 服务器、惠普阿尔法服务器、IBM s/390 大型机、或者轻量级的 ARM Cortex 上网本、惠普 Marvell Orion Media Vault mv2120 存储设备上运行 Linux 操作系统，那么 Debian 能够完全满足你的这一苛刻要求。无论你是升级服务器或者只是想要让整个办公室的电脑有一个共同的接口，Debian 无疑非常方便。

此外，Debian GNU/Linux 可以用多种介质进行安装，比如 DVD、CD、USB 闪存和软驱以及网络。GNOME 是缺省的桌面环境，包含在第一张 CD 中。其他桌面环境—KDE、XFce 或 LXDE—则可以通过两张新的替代 CD 镜像进行安装。同样还提供 Debian GNU/Linux 5.0 的多架构安装 CD 和 DVD，用它们可以从一张光盘上安装多种架构的计算机；同时本发行还提供蓝光光盘，这就让在一张安装盘上提供一个架构的全部软件成为可能。

除常规的安装，Debian GNU/Linux 也可以直接使用而无需安装。这种特殊的镜像也称为 live 镜像，可用于 CD、USB 闪存以及多种形式的网络启动。起步阶段只提供 amd64 和 i386 架构的 Live 镜像。

Debian GNU/Linux 5.0 的安装过程也已经从多个方面进行了改进：在多个其他改进中，重新加入了对从多张 CD 或 DVD 进行安装的支持，某些设备需要的固件可以从可移动介质上加载，还支持通过布莱叶显示器进行安装。安装程序的启动过程同样也得到很多关注：可以在图形界面上选择安装前端和桌面环境，同时也能选择专家或救援模式。Debian GNU/Linux 的安装系统目前已经被翻译成 63 种语言。

总之，Debian 5 是一个稳定、功能强大并且非常有吸引力的操作系统，我想一旦用过之后你一定会喜欢上它。

发行版发布：VectorLinux 6.0

Linux 发行版 VectorLinux 宣布正式推出 6.0 版（代号 Voyager）。

新版的主要特性包括新的稳定的 GUI 安装程序和超过 1000 个软件包的仓库；桌面是基于 Xfce-4.4.3，有定制的主题和图像；可选 LXDE 桌面。软件方面包括 DVD 播放器、视频音频解码器，

多媒体和 Java 插件，图形软件 Gimp、Inkscape、GtKam、Gqview 和 Xpdf；Firefox、Opera 等浏览器；即时聊天工具 aMsn、Pidgin 和 Xchat；播放器 Mplayer、XMMS、Xine 和 VLC；办公软件 Gnumeric、Abiword 和 Scribus。新版改进了无线网络、安装、可用性和硬件支持。豪华版包含了 KDE-4.2、E17、OpenOffice、Skype、Realplayer、Playonlinux 等软件。

Mozilla 在 SCALE 上展示 Firefox3.1 的新特性

在上周六的南部加州 Linux 展会 (SCALE) 上，Mozilla 的“传道者”Chris Blizzard 谈论了下一版本 Firefox 一些新特性。Mozilla 当前正着手准备正式发布 Firefox3.1，该版本将给 Web 开发者带来更强大的性能，一些实验性的新功能也将在 3.1 中出现。

在 firefox3.1 中，将包含原生的 JSON 解析器，初步的测试证明了 Firefox 中原生的 JSON 解析器将带来性能上的提升，该项特性将被程序开发者广泛使用，因为微软也打算在 IE8 中提供完整的兼容。

另外一项令人印象深刻的特性是，Web 程序开发者可以有效利用 Firefox3.1 里提供的工作线程 (Worker threads) 支持，它将在 Javascript 中支持并行执行 (concurrent execution)。工作线程可以使得它能在后台执行复杂的计算，所以不会使得浏览器和 Web 程序假死或者无响应。

HTML5 视频元素也将抵达 Firefox3.1，它将允许视频内容通过 Javascript 直接内嵌到页面，并通过 DOM 操作。Firefox3.1 经内置支持 Ogg Vorbis 和 Theora 格式——开源音频和视频编解码器不会受到专利的困扰。这些编解码器将通过浏览器本身执行，所以该格式内容都可以直接播放无需安装扩展和插件。另外他还在本次展示中谈到一些新特性，包括 HTML 中的 SVG 过滤器，跨站点 XML HttpRequest、DNS 预取和内置字体支持等。



MySQL 创始人正面回应离职原因

近日，MySQL 创始人之一迈克尔·维德纽斯 (Michael Widenius) 正式宣布从 Sun 公司辞职，并在其博客中解释了离开的原因，以及他将来所专注的工作，以下为其全文：

我现在已经离开 Sun，加入我自己的公司 Monty Program Ab。

在去年 8、9 月份曾经有许多关于我辞职的传闻。当时我不希望对这些传闻做出回应，因为我依然在尝试与 Sun 共同解决一些事情。现在我终于可以和大家聊聊所发生的事情了。

这些传言并不是空穴来风，它们有一些事实因素。当时我曾经向管理层提出辞职，因为我坚信 MySQL5.1 还没有做好发布的准备，在其正式 GA 之前，还有许多问题有待解决。我的坚持和其它人的努力的确取得了预期的效果，我与 Sun 管理层达成了一个协议，不立即辞职，而是继续待 3 个月，帮助 Sun 解决 MySQL 开发中的一些事情，也让 Sun 利用这段时间为我创建一个最佳的角色。但是，现在已经过去了六个多月，我所期望的变革并没有足够快的发生，Sun 并没有调整其 MySQL

数据库部门来修复其开发和社区问题。最后，Sun 和我都认为，离开 Sun 公司我才更可能实现我的目标，因此不得不接受这个结果。我们算是好聚好散，双方都期望继续进行业务和工作的合作。

离开的主要原因是，我对 MySQL 服务器版的开发方式不满意，这一点大家从我以前的博客文章中也可以看出来。尤其是，我一直希望它的开发能够在一个真正开放的开发环境中进行，这样才会鼓励来自外部开发者的参与，消除源代码上的差别。尽管 Sun 公司也一直在考虑开放 MySQL 服务器版的开发，但是其步伐显然太慢。

即使到今天，我依然认为，对于 MySQL 来说，Sun 是最合适的买家，未能共同把事情解决，我对此感到非常难过。Sun 公司正在进行很多非常好的工作，我希望它们将继续创建和推动开源的发展。我依然会向 Sun 公司提供帮助，帮助它们实现在开源领域的目标。

下面再来谈谈我的新公司 Monty Program Ab。

Monty Program Ab 将是一个真正的开源公司，它的规模非常小，员工数量保持在 10 到 30 人之间，每个人都是公司的主人，在这儿我们重视员工，并且愿意在一起合作和共享一起创造的利润。你可以从以下网址中了解更多该公司的信息：<http://zak.greant.com/hacking-business-models>。部分 Maria 研发团队的成员将会和我一起离开 Sun 公司，计划继续进行 Maria 项目的研发。和以前最大的区别是，Maria 项目和它的邮件列表将被移至 Launchpad 托管平台上，而且我们将开始使用自由节点(free-node)来进行我们的 IRC 交流(频道#maria)。Maria1.5 现在正处于 beta 版阶段，我们预计不久将推出其二进制版。我们已经开始研究 Maria2.0 的功能和性能问题。由于我的原因，Maria 的开发可能会被耽误一小段时间(大约 1 到 2 个月)，因为我必须处理新公司的一些事情。



新公司将积极参与到 MySQL 社区中，实现对 MySQL-5.1-Maria 分子的稳定修补，并围绕这个分支创建一个活跃的社区。我们还将进行 MySQL 代码的一些架构调整，以使其更简单、快速和更少漏洞。MySQL-Maria 不应被看做 MySQL 的一个分裂，而应该是一个分支，因为我们会将所有官方版 MySQL 的修改加入到 MySQL-Maria 中；其中有些修改可能会被重新完善，但是我们将尽我们最大的努力来确保，终端用户不会感觉到它们的差别。新公司还将在 MySQL 和 Maria 上对客户进行非重发性设计(Non-recurring engineering, NRE)，并将这个工作放到 MySQL-Maria 中。另外，对那些从事开源和社区产品开发的创新技术公司，我将继续提供投资。

另外，我还将开放一个新型饭店，它使用数据库来获得更好的客户体验。(对数据库开发者提供特殊折扣!)。更多信息以后再聊。

深入理解 ubuntu 不仅仅再是桌面系统

一项由 Canonical 和分析公司 RedMonk 正在进行的调查指出：Ubuntu 不应该仅仅被看做是 linux 桌面系统了。

在最近的调查中,大概 7000 多个企业正在使用着 ubuntu 系统,该份报告显示,用户主要利用 Ubuntu 执行诸如网站、数据库、档案、打印及邮件服务器等一般任务,同时也有用户利用 Ubuntu 作为安全、备份及防火墙等重要平台。

Ubuntu 以及它的商业赞助商 Canonical 公司,共同发起了调查,并在博客中发布了结果。被调查的企业来自世界各地,但主要是北美和欧洲市场。

据调查,这些 Ubuntu 用户的企业使用最多的服务器操作系统仍为 Windows 版本, Linux 版本居次,而 UNIX 则排名第三;在 Linux 版本中,使用 Ubuntu 的比例最高,其他依序是红帽的 Enterprise、Debian、Centos、SuSE Linux Enterprise 及 Fedora 等。接受调查的用户企业大部份座落于欧洲及北美,以科技服务业占多数,其次是软件业及教育机构。

至于 Ubuntu 服务器版本的用途排名第一的为网络服务器或网络应用程序服务器,其他较常见的用途还包括文件服务器、数据库服务器、电子邮件服务器、备份服务器、DNS 服务器、开发或测试服务器、安全服务器及虚拟化服务器等。

Ubuntu 衍生于 Debian Linux 操作系统,是在 2004 年首次推出,在短短的几年时间内,已成为最流行的非商业 Linux 发行版。Ubuntu 桌面系统的普及是受到 Dell 公司的启发(Dell 机器的系统上预装 ubuntu),可以肯定的是,并没有威胁到 Windows 的市场份额。Ubuntu 的普及推动了更多对 netbooks 感兴趣的用户,或迷你笔记本电脑,因为他们普遍被装在这些机器上。

Linux 基金会: 已做好反击微软准备

3 月 1 日消息, Linux 基金会(the Linux Foundation)郑重声明,假如微软对荷兰汽车导航系统开发商 TomTom 的诉讼涉及 Linux 内核的话,该基金会将随时与微软“开战”。

据国外媒体报道,微软上周三宣布,该公司已向美国一家地区联邦法院和美国国际贸易委员提起诉讼,指控 TomTom 侵犯了微软 8 项专利,其中 5 项与车内导航技术有关,另外 3 项涉及文件管理技术。Tomtom 的 GPS 设备采用的是 Linux 系统,尽管微软声称 Linux 侵犯其专利期已有多年,但该案被视为微软状告 linux 侵权的第一案。



2007 年,微软曾公开表示 Linux 侵犯了微软 235 个专利。之后,微软便签下了好几个专利合约,最著名是跟 Novell 的部分,可保护用户不受专利提告。微软一年前也曾公开宣示不会针对个别开发者提起诉讼,但对于使用 Linux 的公司则会保留追诉权。

Linux 基金会执行主任吉姆-泽林(Jim Zemlin)呼吁相关各方“保持冷静”。此前,微软首席代理律师助理表示,微软的专利侵权诉讼只针对 TomTom 的 GPS 图形软件,而不是 Linux。

不过泽林也向开源社区承诺, Linux 基金会和开放创新网络(Open Invention* Network)拥有充足的资金,必要时可积极应诉以保护 Linux 的利益。TomTom 的 GPS 图形软件采用了 Linux 内核,以下是 Jim Zemlin 博客的原文。

保持冷静

眼下微软与 Tom Tom 的诉讼仅限于两家公司之间就 GPS 图形软件的专利权产生的分歧，尚难以就该诉讼对 Linux 相关技术造成的影响过早下结论，因此大家要保持冷静。任何专利诉讼在声明专利权被侵犯，到最后查明专利是否有效及侵权事实是否成立，这中间一般都需要一段很长的时间。不管如何，只要该诉讼涉及 Linux（事实上，微软副总裁、知识产权与许可事物首席代理律师代助理霍拉西奥-古铁雷斯已声明，Linux 并非诉讼目标），相信 Linux 相关实体拥有充足的资源来积极应诉。

最好的结局

我们真诚希望微软能够认识到，类似的诉讼只能成为整个软件产业的负担，不符合双方客户的最大利益。相信客户并不希望看到无休止的诉讼，他们的最大愿望是，软件公司能够专注于创新产品的开发。

最坏的打算

Linux 基金会与其伙伴开放创新网络密切合作，我们的所有成员都最好了积极应诉保护 Linux 的准备，我们对基金会的能力充满信心。事实上，类似诉讼是我们的商业模式和法制体系不完善的副产品。我们将密切关注形式的发展，随时做好发起反击的准备。

周四，开源界领导人纷纷发言，认为微软的这一行动是在向开源界挑衅。软件自由法律中心组织(SFLC)负责人爱本-摩尔根(Eben Moglen)表示：“微软的行动就是在发出威胁，倡导软件自由的人们应该奋起自卫，我们的组织正在对形势进行评估，以便采取对应措施。”摩尔根还认为微软的此次鲁莽举动抵销了其近年来在与开源界加强合作方面的努力，使其一直宣称要以“和平方式与开源界共同努力”的言论受到了质疑。

著名开源组织 Samba 的领导人杰里米-爱里森(Jeremy Allison)也在一封电邮中表示：“这实在太令人意外了，微软居然会以为他们在这么做的前提下，还能继续与自由/开放源码软件开发界保持良好关系。现在就让我们对微软天鹅绒手套后面藏着的强硬手腕进行还击吧。”

Novell 裁掉部分 openSUSE Linux 开发者



上周微软和红帽刚达成跨平台虚拟化合作协议，今天 Red Hat 的竞争对手 Novell 却裁掉了未知数量的 openSUSE 社区的雇员。大家都知道这个社区出来包括自愿效劳的开发者之外，还有 Novell 的雇员。在本月初 Novell 也进行了 100 人的裁员。

据被解雇的 openSUSE 社区成员称，“最近 Novell 进行裁员是由于当前的经济状况，但是很不幸，此举影响到了 openSUSE 社区的一些成员，他们在本周初期被裁掉了”。

Ubuntu 9.10 的开发代号被定为“Karmic Koala”

Ubuntu 创始人 Shuttleworth 已经宣布了 Ubuntu 9.10 的开发代号—Karmic Koala。Ubuntu 9.10 为继 Ubuntu 9.04 之后的版本，计划今年 10 月发布。

Shuttleworth 表示，Ubuntu 9.10 有可能会采用新的 Splashscreen，系统引导时间将继续改善。对于适用于上网本的 Ubuntu Netbook Edition 而言，除了更新来自 Moblin 的所有新技术之外，也将对其界面作进一步的打磨。在服务器版本方面，Ubuntu 9.10 则将对 Amazon 的 EC2 API 提供支

持，同时将集成 Eucalyptus 项目，以允许用户主机基于云计算的应用。

BASH 4.0 发布

作为多数 Linux 系统及 OS X v10.4 默认的 shell，Bourne-Again SHell 4.0 正式发布。BASH 4.0 修改了 3.x 版本中留下的 bug，同时增加了许多新特性。

最值得关注的新特性是关联数组（associative arrays），改进了可编程完成函数（completion functionality），case-modifying 单词扩充，复合进程，支持`**'特别 glob 模式，增加 shell 句法和重定向。shell 在分析内部命令替换上变得更严格，修正了 POSIX 虚拟接口的不兼容块。你现在可以去下载源代码，自己去编译它。

开源移动应用：谷歌苹果不相上下

虽然 Google Android 平台并未获主流青睐，但却捕获了不少开源开发者的芳心。Black Duck Software 的新报告显示，移动平台的开源项目，Google 仅落后于苹果少许。

Black Duck 从 4000 个网站中收集了超过 185,000 个开源项目，其中大约 2300 个是针对移动平台的开源项目。尽管数量并不算多，但 2008 年的增长却极为迅速。08 年苹果 iPhone 有 266 个开源项目发布，其次是 Android 的 191 个。从总数上看，Palm 平台仍旧遥遥领先，共有 1,850 个开源项目，但 08 年发布了 113 个，呈现颓势。iPhone 共有 391 个，Windows Mobile 有 359 个（08 年新增 174 个），Symbian 有 322 个（64），Android 有 246 个，黑莓 237 个（96），Maemo 有 56 个（17），LiMo 有 28 个（6）。

行业观察

红帽重拳出击虚拟化 欲重分市场版图

在经济环境不确定的今天，虚拟化就像一棵救命稻草。这棵“稻草”不仅对于厂商来说意义非凡，对于 IT 预算面临紧缩困境的用户也同样充满诱惑力。

2008 年，虚拟化市场显得非同寻常的热闹，VMware 的 ESX 和 VI3、微软的 Hyper-V、思杰的 XenServer、Oracle 的 VM、Sun 的 xVM……加上英特尔和 AMD 等芯片厂商以及众多服务器厂商的大力鼓噪，虚拟化可谓风光无限。也难怪，神奇的虚拟化技术不仅能提高系统资源的利用率，还能带来一系列高效、灵活、自动化的 IT 管理工具，用户可以省成本，提效率，厂商可以找到新的商机，大家何乐而不为。然而，虚拟化技术毕竟不是灵丹妙药，这项技术本身还处在发展过程当中，随着用户越来越多，一些不足之处也开始浮出水面，而这些也成为后来者挑战市场领导厂商的杀手锏。



虽然目前虚拟化已经被公认为是一项主流技术，但虚拟机在全部数据中心服务器中的比例还不足 20%。大多数 x86 虚拟化方案通常只能部署在非关键应用环境中，或者只是用来做做开发和测试。原因自然是多方面的，根据 IDC 在 2008 年进行的一项调查，发现阻碍用户部署虚拟化的因素主要在于：用户不了解虚拟化技术，缺乏有能力实施和管理虚拟环境的技术人才，虚拟化软件太贵，初期投资成本太高，同时对于虚拟化能否真正适应自己的应用环境充满疑虑。

2 月 23 日，在全球 Linux 市场中占据 80% 份额的开源操作系统领导厂商红帽宣布了该公司在 2009 年的虚拟化策略和路线图。红帽大中国区总裁胡伯林在会上表示，红帽在未来 3-18 个月里面将提供包括“红帽企业 Linux、红帽企业虚拟化 Hypervisor、红帽企业服务器虚拟化管理、红帽企业桌面虚拟化管理”在内的一系列新产品，为用户提供从服务器到桌面的全面虚拟化，主要针对那些需要大规模部署虚拟机的企业级用户。

在红帽看来，虽然今天许多专有虚拟化方案带有强大的管理工具，能够减轻部署及管理负担，但仍然受到许多制约，比如性能不够好，可扩展性有限，稳定性和安全性不足以及高高在上的成本。

红帽在其一份公开的白皮书中宣称，红帽企业虚拟化平台将“提供经济高效、易于管理的解决方案，同时决不牺牲性能、可扩展性或安全性。”在 VMware 占据虚拟化市场领导地位、微软 Hyper-V 凭借强大品牌营销实力开拓虚拟化版图的今天，作为挑战者的红帽把焦点聚集到了现有产

品的弱点和不足之上。

性能与扩展

胡伯林告诉记者，传统的虚拟化产品依赖于硬件仿真，首先要拦截来自客户虚拟机的硬件请求，然后把这些请求翻译给主机服务器，从而带来了较大的资源开销，尤其是对于数据库及企业应用等 I/O 密集性工作负载更是如此，这些开销使性能下降 40% 甚至更多。这也使得数据库等企业级生产应用很少进行虚拟化。另外，许多虚拟化产品普遍存在的“时钟漂移”现象也迫使那些运行低延迟应用或时间敏感型应用（如金融交易系统）的企业仍然选择把这些应用部署在传统的物理服务器上。

除了性能，扩展性也需要面临突破。目前这一代 X86 虚拟化软件通常把客户机限制为 4 个虚拟 CPU，使得企业无法对需要占用大量 CPU 资源的 ERP 类企业级应用进行虚拟化。

其实，红帽一直在寻找虚拟化市场的突破口。其企业 Linux5 自从 2007 年初发布以来就一直包含 Xen Hypervisor 技术，在操作系统中集成了虚拟化功能，目前已经在联通 IDC、上海宝钢等地实现了成功部署。但随着思杰收购 XenSource、Novell 集成 Xen 虚拟化模块以来，红帽再想通过 Xen 翻盘市场格局已经很难。于是，当两年前，几个以色列人创办的小企业 Qumranet 发现了一种新的虚拟化技术——KVM（基于内核的虚拟机管理，Kernel Virtual Machine）时，红帽瞅准了时机。2008 年 9 月，红帽正式收购了这家公司，包括 KVM（内核虚拟机）平台、SolidICE 以及虚拟桌面架构（VDI），为红帽推出新一代虚拟化平台奠定了基础。

据称，基于红帽企业 Linux 内核和 KVM 技术的红帽企业虚拟化 Hypervisor，是一种新的、轻型、占用内存少、易于管理的独立 Hypervisor，它可以支持高达 96 核及 1TB 内存的硬件主机，同时上面的虚拟机也可以实现多达 16 个虚拟 CPU 和 64GB 内存的扩展，并且可以同时支持 Linux 和微软 Windows 平台。由于 KVM 在 2006 年就已经被写入 Linux 内核，成为 Linux 开源社区内核小组官方认可的虚拟化软件，因此，它被认为是 Xen 在开源领域的强劲对手。

这也意味着，红帽企业 Linux 的下一个版本 5.4 将在虚拟化方面实现“两条腿走路”，既有过去的 Xen Hypervisor，也有全新的 KVM Hypervisor。胡伯林告诉记者，这两种 Hypervisor 都将得到红帽的完全技术支持，现有的客户可以继续使用 Xen，并在合适的时候迁移到 KVM，红帽承诺仍在红帽企业 Linux5 的完整生命周期内支持 Xen，预计至少延续至 2014 年；而新的客户可以一开始就直接使用技术更先进的 KVM。据悉，红帽企业 Linux 5.4 计划于今年的第三季度推向市场。

看来，KVM 将成为未来红帽虚拟化产品的基础。

安全与稳定

安全性和稳定性也是令用户担忧的主要问题。当企业把一台物理服务器划分成多个虚拟机之后，实际上是把所有的鸡蛋放到一个篮子中。显然，这个“篮子”需要足够的牢固可靠。毕竟，如果下面的物理平台崩溃，或者受到病毒侵扰或黑客攻击，那么，所有的虚拟机都可能会被殃及。另外，虚拟机之间的安全隔离、虚拟机从损坏硬件平台向新平台动态迁移的功能也同样重要。这些其实都已经得到了 VMware 等厂商的高度重视。

跟其他厂商的做法不同，红帽企业虚拟化不是简单地在 Hypervisor 之上再添加一层安全性产品，而是在 Hypervisor 内核中就包含了安全组件：SELinux。跟其他厂商的做法不同，红帽企业虚拟化不是简单地在 Hypervisor 之上再添加一层安全性产品，而是在 Hypervisor 内核中就包含了安全组件：SELinux。SELinux 是与美国国家安全局（NSA）合作开发的安全增强型 Linux，可以在 Hypervisor 中提供军事级别的安全性，让企业添加安全性策略以及强制的访问控制。

管理工具

作为面向企业级应用环境的虚拟化方案，离开了强大的管理工具是难以想象的。特别是对于需要大规模部署虚拟机的用户来说，管理工具可谓至关重要。过去，业界围绕“服务器蔓延”的困扰采取了大量的努力，而随着虚拟化的普及，“虚拟机蔓延”的问题也会越来越突出。

虽然目前多数虚拟化方案提供的分层的工具或许可以管理数百台虚拟机，但用于更大规模的部署时，它们就崩溃了。如果虚拟化打算超越在开发及测试环境中有限的部署，而成为生产系统中完整规模的部署，那就需要搜索驱动、能够扩展至数千台虚拟机的管理能力，具体包括：可以动态监控成百上千套虚拟机的运行状态、资源利用率和负载情况；可以方便地备份和恢复虚拟机的工作；可以把发生故障的主机上的虚拟机自动地在其它主机上重启，避免停机；可以在不停机且不影响业务正常运行的情况下把运行中的虚拟机从一台主机迁移到另一台主机；可以根据资源利用率和定制策略对虚拟机进行负载均衡；可以在非高峰时段把虚拟机集中到更少的服务器上，以节省电力消耗；可以避免计划内维护窗口期间虚拟机的停机等等。

据介绍，新一代红帽虚拟化产品将包括服务器虚拟化管理和桌面虚拟化管理。

胡伯林介绍说，前者计划成为第一套在虚拟服务器和桌面之间实现完整集成管理的产品，其简单、可扩展的基于 WEB 的用户界面和脚本编写接口以及检索驱动的管理能力将使管理数千台虚拟机的工具变得非常容易，并将支持一系列高级管理功能，如图像管理、现场迁移、高可用性、系统调度、电力节省、维护管理及基础架构监控和报告。

桌面虚拟化管理则将为虚拟桌面提供远程提交技术及一套 Hypervisor 栈，将包含一套连接中介、供应管理、图像管理、高可用性及其他功能。值得一提的是，SPICE 远程提交技术能够在虚拟机上为用户提供与物理 PC 毫无差别的体验。

成本

天上掉不了馅饼。部署虚拟化虽然可以减少硬件采购成本，节省人力和电力费用，但虚拟化软件、咨询实施和培训服务仍然需要用户在这方面投入较大的前期投资，尤其是对于企业级用户来说。

红帽认为，尽管目前一些厂商提供了免费的或者非常廉价的虚拟化软件，但却在可扩展性、管理、技术支持方面有限制；而一般的专有虚拟化方案的总成本（含管理工具）仍然高达每台主机 5000-10000 美元。这使得只有那些负载轻、整合程度高（每台主机整合 10-20 台虚拟机）的用户能够获得较可观的 TCO 和投资回报，而现实中用户的需求和应用场合是多种多样的。这使得虚拟化在企业级应用中的普及受到了一定的局限。

胡伯林告诉记者，由于目前红帽全新的虚拟化产品还没有正式发布，还没有正式报价可以公开，但他表示，作为一种开源的产品，红帽企业级虚拟化的售价一定要比目前市场均价低得多，这也是开源的一惯策略。

生态系统

任何一款软件脱离了上下游软硬件生态系统的支持，要想在市场上获得成功几乎是不可能的，虚拟化也同样如此。胡伯林表示，红帽广泛的 ISV 生态系统已经实现“准虚拟化”，也就是说在红帽企业 Linux 上测试和认证的各种行业应用软件可以不需要修改就可获得红帽虚拟平台的认证，这无疑会大大的拓宽客户选择的灵活性。

据介绍，红帽 ISV 合作伙伴项目包括超过 2200 家厂商和超过 3000 个应用方案。由于 KVM 已经集成在 Linux 内核中，红帽企业 Linux 和 Hypervisor 可以实现非常紧密的结合，从而使得红帽的 ISV

合作伙伴能够通过一个应用二进制接口（ABI）来保证他们的认证应用系统在红帽虚拟技术下继续使。红帽会确保处于内核层的 ABI 不会因为大平台版本的发布而改变，使软件和设备驱动也可毫无改变地继续运行。

另外值得一提的是，日前红帽和老对手微软公司达成合作。红帽将参与微软的“服务器虚拟化验证计划”，而微软则参加红帽的“虚拟化认证计划”，这意味着客户能够放心地部署在微软和红帽 hypervisor 上虚拟的微软 Windows Server 和红帽企业 Linux，将得到两家公司的技术支持。不过，该协议没有涉及任何专利和其他开放源代码许可内容，也不包含任何财务条款。目前两家公司的验证都正在进行，首批结果预计将今年晚些时候公布。

红帽虚拟化的市场空间

日前，Gartner 发布了关于虚拟化的最新趋势报告，预测整个服务器和桌面系统虚拟化市场销售额今年将增长 43%，实现产值 27 亿美元，比 2008 年 19 亿美元的市场销售总额有所上升。到 2013 年，大约有 60% 的服务器工作负载将实现虚拟化，只需物理服务器总量的 10% 就可以托管运行，同时将有 10% 到 15% 的企业台式机实现虚拟化。虚拟化市场前景一片光明。

不过，VMware 仍然是今天虚拟化市场上的领导者。根据 IDC 去年发布的 2008 年第二季度全球服务器虚拟化跟踪报告，VMware 的收入份额为 78%，比 2007 年同期的 86% 略有下滑，出货量也从 51% 下降至 44%；其次是 Parallels，收入份额占 16%；微软 Hyper-V 与 Virtual Server 2005 虚拟化软件虽然收入份额仅占微不足道的 1.1%，但出货量份额却高达 23%，比去年的 20% 和 08 年第一季度的 18% 都要高。可见，VMware 正在受到微软等厂商的挑战。而留给红帽的市场空间又有多少呢？

胡伯林没有给出具体的数字，但他相信红帽新一代基于 KVM Hypervisor 技术的开源虚拟化，在性能、可扩展性、安全、管理、成本等方面都占据一定优势。他告诉记者，开源正在成为今天企业 IT 建设的主流——Linux 在整个服务器市场中的份额已经占到 20%，红帽在 Linux 市场中的份额又高达 80%，Apache 占 WEB 服务器的 50%，JBoss 占中间件的 30%，未来开源虚拟化能占到多大的份额，这里有着很大的想像空间？只不过，现在谁也不知道答案！

附：红帽虚拟化大事记

06 年 4 月份，红帽正式宣布了其“集成虚拟化”战略，即在操作系统中紧密集成虚拟化功能。红帽认为：集成虚拟化将成为降低企业成本进程中的重要里程碑。08 年，红帽更是在虚拟化方面动作频频。

08 年 3 月份，红帽最近发布了最新的 RHEL 5.2beta 版，与老版本相比，主要是在桌面和虚拟化方面进行了重大升级。

08 年 6 月 18 日，红帽针对整个企业利用下一代虚拟化技术推出三项战略举措。嵌入式 Linux Hypervisor – 轻型、可嵌入的管理程序，用于托管虚拟化的红帽企业 Linux 和 Microsoft “Windows” 环境；虚拟基础架构管理 – 易于使用的多系统管理对于虚拟化系统在生产环境中的部署至关重要；安全基础架构 – 生产环境中的虚拟化极为关注企业 IT 的安全性。

在 08 年 6 月份举办的 Red Hat 峰会上，红帽宣布了基于 KVM 的新一代虚拟化管理器。该管理器提供的虚拟化方案可包括 Linux 的一切优点——卓越的安全性、高性能和广泛的硬件支持——所有这些只需消耗很少的内存并可以方便地嵌入到服务器和桌面计算机中。

2008 年 9 月 5 日，红帽宣布收购 Qumranet 公司。该收购内容包括 Qumranet 的虚拟化方案，即 KVM（内核虚拟机）平台、SolidICE 以及虚拟桌面架构（VDI），将为企业客户提供一整套的虚

拟平台。根据协议，红帽向 Qumranet 公司支付了大约 1.07 亿美元现金，预计在下一年可增加 2 千万美元的收入。

2009 年 2 月 16 日，红帽与微软首度合作增强虚拟化互操作性。红帽将验证在红帽企业虚拟技术上 Windows 服务器虚拟机并获得技术支持，微软将验证在 Windows Server Hyper-V 和微软 Hyper-V 服务器上的红帽企业 Linux 服务器虚拟机并获得技术支持。

俄罗斯是怎样发展 Linux 的？

作者：袁萌

俄罗斯是我们的友好邻邦，近年来，Linux 在俄罗斯发展很快。有什么经验值得我们学习？为什么？

俄罗斯政府明白一点：Linux 本身并没有国界，可以合理加以利用。去年年底，俄罗斯政府与（俄罗斯）ALT Linux 公司（以及 Armada 公司）签订一份合同（Contract），要求（他们）为俄罗斯学校以及教育机构开发一套“School Server”（所谓“学校服务器”）发行版。这里请注意一点：这是一份合同，就是政府拿钱，买软件（特定的产品），合同所产生的产品，版权归 ALT 公司，但是，全国无偿使用。结果怎么样呢？今年 2 月 21 日，ALT 公司给出一份答案：（俄罗斯）学校服务器发行版（请见 ALT Linux 4.1.0“School Server”一文），并且给出了供全俄罗斯公民自由下载（相关.iso 文件）的 FTP 网站的域名，前后时间只用了 2~3 个月。一手交钱，一手交货，两不相欠，问题就是这么简单。俄罗斯“学校服务器”发行版有什么特性呢？最明显的特性就是，向用户交代清楚使用了什么样的自由软件包，例如：软件包名称.版本号（xx.xx.xx），具体地说，Linux 内核 2.6.25，C 库 glibc 2.5.1，DHCP 服务器，DNS 服务器 Bind 9.3.5，proxy 服务器 Squid 2.6，Samba 服务器 3.0.30，.....，FTP 服务器 proFTPd 1.3.0，打印服务器 CUPS 1.3.9，等等，共计 20 余项。俄罗斯学校服务器发行版的最大特征是：提供了 FTP 服务器、Mail 服务器、Samba 服务器、Moodle 服务器和 Media 维基服务器以及本地语言的系统管理中心。

ALT 公司要保证“学校服务器”发行版可以自由使用，不会发生法律风险，同时，（向学校）提供技术支持服务（政府买单）。最明显的事实是，俄罗斯政府花费了很少的钱（绝对不会上亿卢布），就把问题解决了。自由软件不怕用（消费），使用的越多，效果就越好。俄罗斯是 Linux 的纯粹消费者吗？不是。俄罗斯向世界贡献了“学校服务器”发行版，（全球）任何人（包括任何国家）都可以自由使用。结合我国情况，我们要把全球 Linux 资源先转化为“国产”软件，然后再加以商业推广，企业费劲，用户受累。我们要承认，我们某些企业利用 Linux 资源发了财（虽然不多），而老百姓并未因此得利。我们不知道这个“歪理”是从何而来。最近，古巴政府发展 Nova Linux 发行版的思路与俄罗斯的路子一个样。印度人搞 10 美元计算机绝对不是一桩“搞笑”之事，虽然印度人巧妙使用 Linux 的招数，弄得全世界人直发笑。穷不是一件坏事。Linux 就是治穷的办法之一。

Android 与 Symbian 手机平台之争

《商业周刊》网络版近日撰文称，谷歌和 Symbian 协会开放手机操作系统平台并不意味着手机行业的和平和谐，因为这两家采用不同“开放”手机系统平台的竞争对手，已经开始互相攻击。

在开放源码软件的情况下开发下一代手机操作系统的斗争中，谷歌和 Symbian 都表示能够开发出更高级的操作系统。如果你认为开放手机操作系统平台意味着手机行业的和平和谐，那么你就大错特错了。因为谷歌和 Symbian 这两家采用不同开放手机系统平台的竞争对手，已开始互相攻击。

谷歌两年前公布手机系统平台 Android 后挺进手机市场。Symbian 协会也正在将不同的 Symbian 手机系统整合成统一的平台，并将于今年晚些时候发布统一的 Symbian 平台。尽管谷歌去年公布了 Android 的源代码，但 Symbian 协会主管李·威廉姆斯(Lee Williams)表示，基于 Linux 的谷歌 Android 平台只是穿着“公开”的外衣，而实质上并未公开，“这只是营销手法，Android 仍处于谷歌控制之下”。威廉姆斯表示，“如果你问我 Symbian 协会未来两年的走向，我将告诉你



我不知道，我能告诉你计划，但 Symbian 协会的走向将由协会会员共同决定”。谷歌移动平台部门主管里奇·米纳尔(Rich Miner)在上周举行的移动通信世界大会(MobileWorldCongress)上对由团队引领的开放平台提出反对意见。米纳尔称，“如果你认为谷歌没有完全开放 Android 平台的源代码，那么这时你对开放源代码的误解”，“如果你需要通过加入一些组织来获得源代码，那么这并不是真正的开放源代码”。Symbian 协会会员可以通过免版税许可获得 Symbian 平台源代码。企业和组织都能加入 Symbian 协会，会员年费为 1500 美元，但该协会不接受个人会员。

米纳尔也对谷歌控制 Android 平台的看法提出反击。他说：“谷歌嵌入 Android 平台的技术都已经开放，我们的竞争对手所知的内容跟我们一样多”，“我们认为有人控制整个平台不利于行业发展，开放平台将产生巨大的杠杆作用，这也是为什么谷歌向开放平台和技术投入了大量资金”。米纳尔表示，谷歌是商业企业，与 Symbian 协会这种非盈利性组织不同，因此谷歌并不“认为所有东西都有必要开放和免费提供”。威廉姆斯认为 Android 将面临的关键挑战是分散化。米纳尔对此并不认同，他认为 Android 有助于联合开放源代码的企业。

MySQL 两高层离职对 Sun 影响有多远

摘要：随着 MySQL 两位高层的离职，有人担心会给 Sun 投资 10 亿美元的数据库计划带来麻烦。但是，事实上也许不会，让时间来证明吧。

这十来天，对 MySQL 来说，可谓是新闻频频。

首先，开源数据库 MySQL 创始人 Monty Widenius 离开了 Sun 公司。一天之后，同样的事情发生了，MySQL 前 CEO Marten Mickos 也决定离开 Sun 公司。

这个消息，对于 Sun 公司拿 10 亿美元投资的 MySQL 会不会造成危害呢？Sun 会不会处于投资失败的风险中呢？就我个人而言，两位高层的离开，Sun 公司的 MySQL 部门处境未必糟糕。MySQL 部门是一个大的部门，并不会因为两个人的离开而处于瘫痪，不管这两位高层曾经的头衔有多高。当然，话说回来，Widenius 作为 MySQL 的创始人，在 MySQL 界内还是占有很特殊的席位的，对吧？

然而，MySQL 的境况也未必会更糟糕。至少从去年开始，Widenius 已经投身于 Maria 引擎(一种 MySQL 数据库的插件)。虽然他会离开，但是他还会继续专注于 Maria 引擎，对 Maria 引擎来说，这并非坏事，因为他的工作不会因为 Sun 公司所打断，或许会取得更好成果。如果在 Sun 公司，

Maria 的开发或许会受到 Sun 公司的束缚。

对于 MySQL 来说，还有其它的事务数据库引擎，知名度很高的 InnoDB 引擎，该引擎自 2005 年以来由 Oracle 公司所拥有。从 2006 年开始，MySQL 也一直在开发自己的事务数据库引擎，名为 Falcon。这个工作不是由 Widenius 来领导的，而是由 Jim Starkey 来充当先锋力量，Starkey 的公司 NetInfrastructure 在 2006 年 2 月份被当时的 MySQL 收购。有人会说对于 MySQL 数据库的发展来看，Starkey 的 Falcon 工作的重要性至少和 Widenius 的 Maria 同样重要。

Widenius 对现在的 MySQL 5.1 版本不满意，并且还不能阻止它。简单的事情就能看出来，Widenius 对 MySQL 的控制力没有以前那么强了，Widenius 一直没有领导 MySQL 主要的大项目。虽然他的建议会造成影响，但实际上管理 MySQL 开发的还是 Sun 基层人员。结果就是 Widenius 的离开并不会对 MySQL 数据库的开发造成直接影响。至于 MySQL 的前 CEO Marten Mickos，他的离开并不是出于意料之外。他的离开会不会对 MySQL 数据库的未来有危害呢？肯定是有的一些的，Mickos 是 MySQL 的业务领导者，但是像 Sun 这么更大的公司，拥有许多经验丰富的执行官来说，Mickos 对 MySQL 的控制就大不如从前了，所以离开也是预料之中的。

从领导能力的观点看，我认为 Sun 自己的 CEO Jonathan Schwartz 对开源有很清楚的认识，并且我毫不怀疑他至少能将 MySQL 管理得和 Mickos 一样好。

一个开源公司的 CEO 在自己的公司被更大的公司收购后离开并不是前所未有的。在 2006 年，Linux 提供商 Red Hat 曾经以 3.5 亿美元收购了开源 Java 中间件提供商 JBoss。仅仅一年之后，也就是 2007 年，JBoss 的创始人 Marc Fleury 离开了 JBoss。Fleury 的离开减少了 JBoss 的销售额了吗？两年后，答案是否定的，它成为 Red Hat 势头上升的业务并且带来收入。

在 IT 界内关于公司创始人和新技术发明者有一些神秘色彩。每个人都知道 Steve Jobs，Apple 公司的创始人，他在被迫离开后又回来继任，直到由于身体原因才离开。另外，开源界有 Linux 创建人 Linus Torvalds，这 10 年来他一直领导 Linux 内核的开发。但是 Jobs 和 torvalds 的案例也不需要成为标准。在公司被收购了，事情就发生改变，虽然创始人可以坚持做原来的工作。只有时间告诉我们在 Widenius 和 Mickos 的离开后，Sun 领导的 MySQL 会不会比他们在位的时候更强大。

但有一件事情是肯定的：MySQL 应用很广泛，部署 MySQL 的很多。由于 Sun 公司的努力，加上开源社区的努力，MySQL 会继续向前发展，存活的会更久。

圆桌论道：经济危机、开源人才

在 2008 年我们看到的是中国的开源界风起云涌，而 2009 年在经济危机的大环境不利因素影响下，中国的开源界又将迎来什么样的光景呢？近期，IT168 有幸邀请到了中国开源业界的风云人物：中国工程院院士倪光南、北京中科红旗软件公司副总裁郑忠源博士、上海中标软件有限公司常务副总裁秦勇、北京红旗 2000 软件技术有限公司总经理胡才勇、Discuz 创始人、康盛创想北京科技有限公司总裁戴志康五位嘉宾，在 Intel 开源战略经理陈绪的主持下就经济危机、开源人才、核高基等话题展开圆桌讨论。

主持人：金融危机对开源企业的影响，或者说带来的契机。

倪光南：现在金融危机对整个 IT 消费对象来讲，肯定是不利的，但是另一方面，开源软件还有一个价格因素，比如说原来是最贵的是最好的，但是现在要考虑降低成本，这时候会考虑到这个成本，虽然我们中国第三位考虑成本，第一位是信息安全，第二位是供货商，到底两个因素一个是消极的，一个是积极的，到底对我们做开源软件的公司是好是坏，目前不敢说，看时间。到下半年怎

么样，我们再来看。

郑忠源：这个金融危机从 08 年美国次贷开始，影响到中国已经下半年的事了，目前在中国的表现主要体现在江浙、广东一带的中小制造业企业，对我们这些企业目前的影响不是特别大，但是这个影响在今年会体现出来，这两个趋势是并存的，哪个会占上风，目前还不好说。一方面如果企业 IT 业务一定要做，但是要控制成本，这个时候我们的机会就来了，但是假如他说这件事情我们可以不做，我 IT 整个停下来，那对不起大家就受灾了。

胡才勇：金融危机在中国对开源的影响不是特别大，因为最大的问题是产业的环境，在这个方面我认为金融危机在这个情况下影响不是很大。但是另一方面对整个产业界，对开源软件投资方面肯定有影响。

秦勇：几位专家都已经发表意见了，我可能有不太一样的观点。我觉得金融危机对中国的影响可能是比较长的一件事情，从 2008 年开始从政府的政策出台，尤其是在基础软件扶持的力度还是很大，最近和未来有希望加大投资力度。毕竟金融危机是从整个银行业开始，慢慢波及其他行业。那么对于中国基础软件来讲，跟银行之间的关系不是那么紧密，所以整体来说金融危机对于中国的基础软件行业，影响并没有大家想象的那么大。我个人在未来三到五年内，应该说对基础软件和这个产业来说，应该说金融危机的影响是可以忽略的。

戴志康：从开源软件的角度来讲，在中国的开源软件企业从来就没有挣过特大的钱，或者真正进入到一线的商业主流里面去，所以既然没有赚到太多的钱，或者有的公司好一点的小康，但肯定不是富人，我们公司就是混个温饱，所以从这个角度来说影响是不大的。

专家专栏

龙芯 GCC 小组：积极参与社区建设

大家好，新的一期专家专栏又和大家见面了，今天有幸邀请到来自中科院计算所的邢明杰工程师，让他来给我们介绍一些他们目前所从事的开源社区事务，他们目前主要负责龙芯 GCC 后端支持的工作。



ChinaUnix: 能先给大家简单介绍一下 GCC 的发展情况么？以及 GNU 的发展情况，因为 GNU 到现在已经 25 年了。

邢明杰: 有关 GNU 的背景和资料，网上有很多，我也不必在这里重复了。感谢 Richard Stallman 为我们竖起的这面“自由”旗帜。我想，也许就是 GNU 的这种自由的精神，才吸引了无数的贡献者，推动着 GNU 一直走到今天。GCC 是 GNU 计划中最早开发出来的软件之一。

我们知道，GCC 最初也是由 Richard Stallman 开发出来的。GCC 的第一个版本于 1987 年发行，只支持 C 语言；现在 GCC 已经支持 C，C++，Objective-C，Fortran，Java，Ada 等语言。GCC 的含义也由最初的 GNU C Compiler 演变为 GNU Compiler Collection。据统计，现在 GCC 拥有的 maintainer 有近四百人，contributer 数千人，tester 数万人。规模可谓壮大。

ChinaUnix: 我听说你之前创建过 HelloGcc Workshop，这是一个什么样的社团，现在的状况如何？

邢明杰: 其实，只是在网上注册了一个空间。当时我正在尝试翻译 GCC Internals，想通过这种方式来强迫自己读一遍 Internals。于是便申请了一个网上空间把翻译放到上面。翻译的很粗略，而且直到现在还剩下了许多章节没有翻译，呵呵。不过，也因此在网上认识了许多朋友。teawater 和我在 2008 年 10 月份举办了一个小型的关于 GCC 和 GDB 的技术讨论会，邀请大家在一起互相交流、沟通，感觉受益匪浅。我也希望今年可以继续举办第二次技术讨论会。这是一个比较随意的社团，主要是针对 GCC 工具链方面的，随时欢迎大家加入。

ChinaUnix: GCC（以及 bintuils）现在已经加入了对龙芯 2E/F 的支持，你对此有何看法？

邢明杰: 首先，我想说明的是，对于这些工作，我并没有做出什么贡献。在 binutils 邮件列表上搜索 loongson 关键字，就可以看到，binutils 社区中与龙芯相关的邮件最早是 2007 年 2 月份从 gentoo.org 发出的。我是 2008 年初才开始接触龙芯，当时注意到 GCC 还没有支持龙芯后端，大家还只是使用 mips 的后端来编译龙芯上的操作系统以及应用程序。而国内已经有一些朋友在网上发

布了自己的 patch。其中我所知道的比较活跃的两位朋友是 Zhang Le 和 jamesr。我便也做了一个简单的移植，并在 GCC 邮件列表上发邮件询问是否有人正在做龙芯 GCC 的移植工作。紧接着 Maxim（来自 codesourcery）便回复我说他们已经着手要提交 patch 了。

但是，从这件事情上，我意识到一个问题。那就是，我们需要融入到社区中，积极参与邮件列表的讨论，提交基于 svn 开发版本的 patch，只有这样才能将自己的工作贡献到最终的 GCC 发行版中。如果只是在国内的论坛上发一些帖子，或者自己基于某个版本做一些修改，搞出一个仅供内部使用的版本来，最终还是无法被社区接受。其实，GCC 有许多分支项目，感兴趣的朋友可以到官方网站上查看一下，上面也介绍了如何提出贡献。另外，GCC 每年都会召开一次大会，会议论文也可以在网站上获得。我们上次就邀请了在 2008 年 GCC Summit 上发表过文章的 yingbo 来参加我们的讨论会。

ChinaUnix: 你现在在计算所的 GCC 小组工作，能给大家介绍一下这个小组的情况么？

邢明杰: 现在，我们在吴老师的带领下，成立了一个小组，主要从事 GCC 方面的研究工作。一方面，为龙芯芯片的后续研发及时提供 GCC 编译器的支持；另一方面，与国际上的研究机构合作，开展一些前沿的科研项目。小组成立时间还不是很长，主要是由员工和在读博士，硕士组成。我们也希望有感兴趣的朋友加入进来。在这里，我也想感谢吴老师对我的指导，彭亮，袁璐洁，邱吉，徐翠萍，冯睿鑫，陈洋，黄元杰等给我的帮助。

ChinaUnix: 既然作为 GCC 的团队，如何和国际上的开源技术人员协作的，有什么值得国内的技术人员学习的地方？

邢明杰: 我个人认为，非常直接有效的方法，就是通过邮件列表。单从讨论技术问题这样角度考虑，我非常喜欢 Mailing List 这样的邮件列表形式，不喜欢 bbs 这样的网站论坛形式。另外，我们现在跟法国 INRIA 合作，他们这些年在 GCC 社区非常活跃，我们可以通过邮件，电话会议，或者互访的形式进行交流。总之，我觉得交流是最重要的环节。

ChinaUnix: GCC 在和商业编译器的对比中，有何不同？

邢明杰: 我想，如果和商业编译器相比的话，最大的不同就应该是，GCC 是非商业的。你可以自由的获得它的源代码，对它进行修改并且发布。而且，现在 GCC 发展很快，性能也在不断提升，甚至可以和一些商业编译器相媲美。另外，GCC 支持众多的体系结构，非常易于移植。我们知道，linux 本身就是通过 GCC 来编译的。所以说，在 linux 下工作的朋友一定对它不陌生。

ChinaUnix: 龙芯 GCC 移植有哪些特点，在龙芯平台上使用 GCC 需要注意哪些问题？

邢明杰: 我们知道，GCC 具有很好的可移植性，而且龙芯又是基于 MIPS 体系结构来设计的，所以龙芯 GCC 的移植相对来说需要改动的地方要少许多。查看一下 GCC 中关于龙芯 2E/F 的代码，就可以看到，主要的工作包括对龙芯特有的多媒体指令的支持，指令流水线的描述以及一些其它新增指令的支持等。对于其它方面，比如说寄存器的使用，函数调用约定等，都是沿用现有的 MIPS 后端代码，这样也就使得在龙芯和 MIPS 具有一定的兼容性。在龙芯 2E/F 机器上，通常如果使用 GCC 缺省的选项来编译程序，就会生成针对 MIPS1 的体系结构的代码。当然，生成的程序也可以执行，但是性能会差一些。如果要生成针对龙芯 2E/F 的体系结构的代码，则可以加上 `-march=loongson2e` 或者 `-march=loongson2f` 这样的选项。

本期推荐

GNU 项目 25 周年，路在何方？

在计算机出现的最初时期，所有的一切都可以被认为是自由软件。当时的计算机体型巨大，不能联网，还有在现在看来非常不能理解的工作方式，因为他们最初都是通过卡片或者纸带来实现计算的。就在这个时候，Richard Stallman 出生了。上个世纪 60 年代的时候，Richard Stallman 开始在 IBM System/360 大型机上进行程序设计，当时采用的语言是 PL/I，这是一个有着悠久历史的过程性编程语言，距今已经有 45 个年头。70 年代左右，Richard Stallman 开始在 MIT 人工智能实验室工作。就在这里，他见证了软件的自由向私有化以及专有方向的发展。正是这种“黑客”文化的不断消失，使得他下定决心要改变这些，从而为他在 83 年创立 GNU 项目打下了基础。



在 Richard Stallman 的记忆里，在整个 70 年代，绝大多数操作系统已经开始走向专有，不再开放。就比如在 MIT 的人工智能实验室，最开始的时候也有一个自由的操作系统，但是在 80 年代早期的时候也已消失。

因为在此时期，MIT 人工智能实验室里的黑客们成立了一家名叫 Symbolics 的公司，这家公司主要是结合当时的软硬件系统，出售名为 LISP 系统的产品。Richard Stallman 认为，这种闭源的方式正是他所追求的自由软件的死对头。

当时 Richard Stallman 花了大量时间和 Symbolics 公司斗争，并且不停的攻击它，而且给那些在 MIT 智能实验室办公的人下了最后檄文。他还曾经说过：“我并不想在我的余生中和谁过不去，我只是想把我们所失去的东西找回来，我只是想使得计算机的使用者能够获得真正的自由。但是，假如程序是专有的，那么这个理想是永远不可能实现的。”

所以在 1984 年的时候，Stallman 决定自己从头开始写一个自由的操作系统。这个操作系统的名字是 GNU，取意递归“GNU is Not Unix”语句的首字母。才开始的时候，他想通过改写一个名为 Patel 的编译器来作为这个操作系统的基础，但是很快发现这样行不通。于是他决定从头开始写一个自由的编译器，这就是后来著名的 GCC（GNU Compiler Collection）的基础。然后，他还重写了 Gosling Emacs 文本编辑器，将其中的 LISP 代码全部剔除，这就是最初的 GNU 项目自由软件项目之一，GNU Emacs。

接着，就是我们见证的一个历史性时刻，因为要发布 GNU Emacs，Stallman 就需要给软件附带一个用户授权协议。最初，Stallman 在 1985 年使用的授权协议并不是 GPL。这是 GNU 项目首次在软件版权领域涉足，在此之后的第四个年头，所有的 GNU 操作系统中部件，所有其他部分都逐渐地采用统一的授权协议，这就是 GPL，GNU 通用公共授权。

GPL 授权协议：生命力很顽强

在第一版本的 GPL 发布二十年之后的今天，我们很难去否认 GPL 授权的影响力和巨大的贡献。目前全球半数的开源软件使用的是 GPL 授权或者 GPL 授权的衍生版本。比如像 GCC、GNU Debugger 和 Glibc 这样的自由软件项目，现在的影响力远远超出了 GNU 操作系统的范畴，在更广的范围内得到应用，起着自由软件运动主力军的作用。

来自纽约法学院的 James Grimmermann 教授认为，之所以 GPL 有如此强大的生命力，就在于它的公开审查和广泛使用的特点。首先，这份授权协议起草的非常完善，虽然很多的开源软件授权协议也非常完善，但是 GPL 要做的更好一些，它通过依靠社区力量，在多年的时候中得到了大量的反馈和审查，其中考虑了很多特殊情况。

二十多年来，GPL 还一直没有真正的出现过版权方面的纠纷，但是 James 教授认为，这并不是妨碍它继续进步的原因。事实上，除了在 GPL V3 版本中，有关 DRM（数字版权管理）的条文还在模棱两可之外，围绕 GPL 本身是授权协议还是合约的问题，一直都是 GPL 争吵不休的主题之一，可以说，GPL 自身还是存在着一定的问题。他还说，GPL 授权和大部分开源授权不冲突，你要是违反了 GPL，你就不可能不违反其他的开源授权协议。

假如没有 Richard Stallman，GPL 授权、GNU 项目，还有自由软件基金会是否会倒下呢？假如 Stallman 退休后，它们将会如何发展呢？FSF 授权咨询工程师 Brett Smith 给了我们明确的答案，他说，FSF 在运作上已经很早就和 Stallman 无关了，FSF 走的是非营利机构的途径。是 GPL 等相关协议的管理者和维护者。在组织上，自由软件基金会会有一个管理委员会，他们负责基金会的发展和运作。

叫 GNU/Linux 还是仅仅叫 Linux?

在 GNU 社区中一个经常吵来吵去的话题就是，Stallman 对于 GNU 项目哲学层面追求的苛刻要求。例如，来自 Apache 基金会的创始人之一，Brian Behlendorf 就曾经说过，stallman 对于某些字眼的过份追求、缺乏对其他自由言论的容忍，着实令人抓狂。

Behlendorf 在某些场合认为，开源软件和自由软件在某些情况下是一样的，而这种说法是 Stallman 旗帜鲜明所反对的。Behlendorf 对于开源项目的社区抱有更加宽广的胸怀，甚至是某些并没有严格遵守 GPL 关于再发布条款的项目。

自从 GPL 发布以来，很多事情已经发生变化，包括软件行业，虽然 GPL 也在发展，但是显然没有跟的上脚步。在 GNU 项目创立之初，GPL 协议不同于那些共享软件，强迫性的反馈制度是必须的，只有这样他才能生存下去，取得发展。Berkeley 授权没有采用这种强迫性的制度，所以没有足够的社区反馈，因而没有得到长足的发展，而这些正是目前 Unix 市场衰落的原因之一。在自由软件基金会创立之初，也只有这样做，他们才能存活下去。

但是现在还抱着这种想法，可能并不是好事情。正如 Behlendorf 所说，Stallman 在 GNU 项目精神上的苛刻追求，已经使得新一代的软件开发人员对他避而远之。自由软件和开源软件两者之间的分歧已经使得很多开发人员感到困惑。Behlendorf 认为，在 Stallman 看来，作为 GNU 项目的创始人，GNU 所追求的软件自由及其开放精神是容不得半点妥协的，但是现在有人在推广自由软件，这伤害了 GNU 以及自由软件运动。而事实上，他本人应该感觉骄傲，因为这也是自由软件运动成果之一。

原文来自：SD Times，作者：Alex Handy

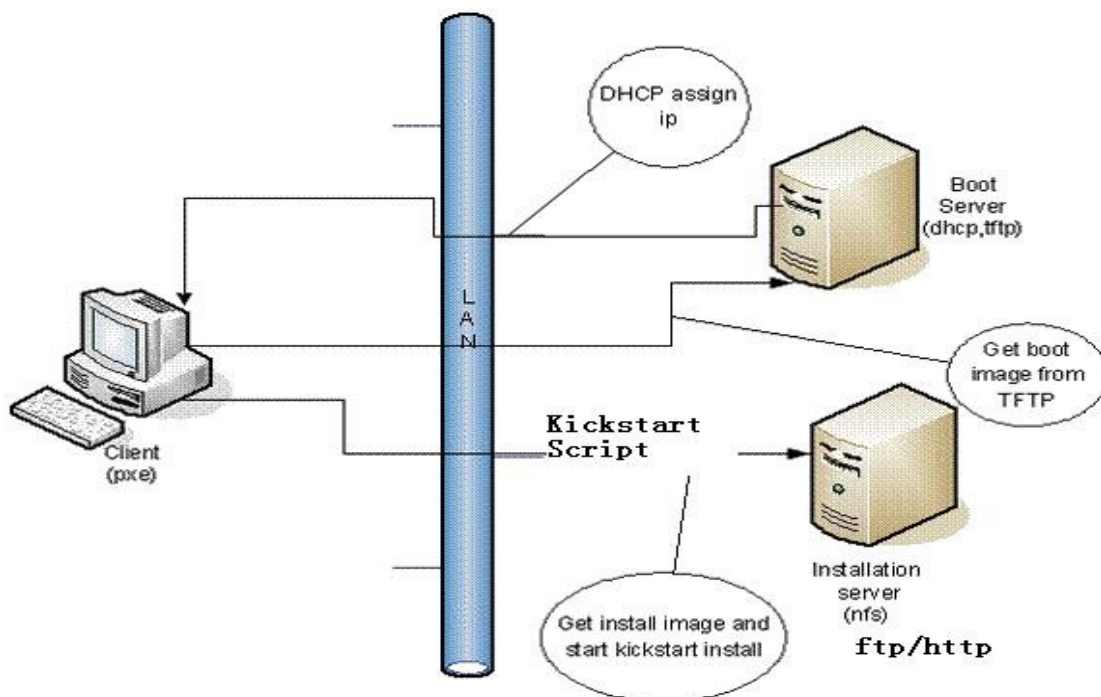
技术新知

独辟蹊径网络安装系列之 Debian/Ubuntu

ChinaUnix 网友: kns1024wh

此文章是 Linux 部署方式系列文章中的第三部分，回归到主流的 Redhat 衍生版体系以及描述对比 PXE 安装服务的实现及使用 Kickstart 在部署主机系统是妙用。本文涉及的内容是对现有 Redhat 体系的网络安装方式的一个总结，重点是在讲述 PXE 安装服务器的核心内容实现方式并不限于 Linux 环境；结合 Kickstart 如何快速的实现安装主机的基本信息的配置，以及如何定制主要软件包，同样灵活应用 Kickstart 和 tftp 可以具体指定那台主机安装那些配置好的参数，节省管理员的时间和精力，自动安装减少了人为出错的几率。好处自然是很多，这里就不过多描述了，下面开始本文的内容。主要针对目前的 RHEL 5.3 版本以及 Centos 5.2 版本为例。

首先熟悉一下 PXE 网络安装服务器的架构。PXE 安装分为两个阶段，第一个阶段是 BOOTP 阶段，此阶段完全由网卡自带的 PXE ROM 芯片所决定，Linux 主机将从 tftpserver 上获得 PXELinux 启动内核和 initrd.img 文件；第二个阶段是在 PXELinux 内核启动 Linux 主机网络安装进程阶段，Linux 主机需要和安装服务器进行网络通信，需要通过 DHCP 服务获得 IP 地址，以便获得安装脚本和获得安装文件。从下图可以看到 PXE 网络安装服务器包含一个 DHCP 和 TFTP 主要解决主机没有操作系统的 BOOTP 阶段获取必要的安装信息，就是通过加载 vmlinuz 和 initrd.img 进入 Linux 安装的引导过程。其次就是要准备一个安装的镜像服务器并通过 nfs、ftp、http 等协议将 Redhat&Centos 的安装源（安装目录树）发布供安装使用，同时简单的变动一下这个安装目录树同样可以成为一个本地的 yum 源。重要的一点就是要提通过 Kickstart Script 配置文件实现安装的自动化。



图：PXE 网络安装服务架构

实际上为了保持这个架构的通用性，将分别讲述在 Redhat&Centos 中的实现方式以及在 Windows 环境中的实现方式。实际上借助一些简单的工具 PXE 网络安装往往可以成为一个便携的方式存在。

在 Redhat&Centos 中安装 tftp 和 dhcp，通过 yum 完成，这个对与 Centos5.2 非常的便利，至于在 redhat 中的操作，可以看后文中描述的推荐的 yum 源的实现。

```
yum -y install tftp-server #完成 tftp server 的安装
yum -y install dhcp*      #完成 dhcp server 的安装
```

实际上 PXE 安装中有疑问的部分就是配置一个 tftp 的目录、设置 dhcp 的 tftp 启动参数、建立 Redhat&Centos 的安装目录树。要清楚这些问题先要从 tftp 主目录/tftpboot 下的目录和文件用途说起。

```
/tftpboot/pxelinux.0 #文件
/tftpboot/*.msg      #文件
/tftpboot/vmlinuz    #文件
/tftpboot/initrd.img #文件
/tftpboot/pxelinux.cfg #目录
```

Redhat&Centos 安装程序的内核 vmlinuz/根文件系统文件 initrd.img 是 PXE 安装过程的第一步，正确引导后将看到 boot.msg (*.msg 中的一个) 提示信息，用户选择安装方式，pxelinux.0 启动镜像文件在执行过程中，会读取配置文件以确定它应该载入什么 Linux 内核文件来运行。所有的配置文件都放在启动服务器的/tftpboot/pxelinux.cfg/目录下。

明白了 tftpboot 目录下的文件和目录的作用就要弄清楚这些文件是从那里获取的，这些文件就在已经安装好的 Redhat&Centos 系统和安装光盘中。为了便于获取这些文件需要建立一个安装的目录树，具体操作也是很简单的 mount 上 DVD 镜像，建议使用 DVD 镜像当然 CD 镜像也是可以的，然后通过如下的命令复制 DVD 镜像上的全部的内容到安装目录树中。

```
mkdir /InstallOs #建立安装目录树文件夹 InstallOs
mount -o loop /home/CentOS-5.2-i386-bin-DVD.iso /mnt #将 centos 的 DVD 镜像 mount
cd /mnt #进入镜像加载目录
tar cf - . | (cd /InstallOs;tar xfp -) #将 DVD 镜像文件全部复制到安装目录树文件夹 InstallOs
```

下面就从安装目录树文件夹和当前的系统中获取 tftpboot 目录中的文件和文件夹

```
cp /usr/lib/syslinux/pxelinux.0 /tftpboot/ #获取 pxelinux.0 文件
cp /InstallOs/isolinux/initrd.img /tftpboot
cp /InstallOs/isolinux/vmlinuz /tftpboot
cp /InstallOs/isolinux/*.Msg /tftpboot

mkdir /tftpboot/pxelinux.cfg #建立 pxelinux.cfg 目录
```

```
cp /InstallOs/isolinux/isolinux.cfg /tftpboot/pxelinux.cfg/  
mv /tftpboot/pxelinux.cfg/isolinux.cfg /tftpboot/pxelinux.cfg/default
```

到此已经建立了一个基础的 tftpboot 目录机构和一个安装目录树。基础工作结束下面进入到 tftp 和 dhcp 的服务参数设置过程。具体设置参考下列配置文件：

1、dhcp 配置文件 cat /etc/dhcpd.conf 中的蓝色部分是设置支持 BOOTP 的参数，注意其中的 filename "pxelinux.0"；如果缺少这个信息逻辑将无法获取到安装程序的内核 vmlinuz/根文件系统文件 initrd.img；红色的部分是 dhcp 客户端获取的 ip 子网的信息，注意保持设置的 subnet 信息的一致性。

```
#cat /etc/dhcpd.conf  
ddns-update-style none;  
ddns-updates off;  
deny client-updates;  
one-lease-per-client false;  
allow bootp;  
option T150 code 150 = string;  
allow booting;  
class "pxeclients"{  
    match if substring(option vendor-class-identifier,0,9) = "PXEClient";  
    filename "pxelinux.0";  
    next-server 172.16.1.9;  
}  
subnet 172.16.0.0 netmask 255.255.0.0 {  
    option routers          172.16.1.9;  
    option subnet-mask      255.255.0.0;  
    option nis-domain       "telecarenet.cn";  
    option domain-name      "lvs.local";  
    option domain-name-servers 172.16.1.9;  
    option time-offset       -18000; # Eastern Standard Time  
    range dynamic-bootp 172.16.1.160 172.16.1.230;  
    default-lease-time 21600;  
    max-lease-time 43200;
```



```
# we want the nameserver to appear at a fixed address
host diskless {
    next-server 172.16.1.9;
    hardware ethernet 00:02:b3:ee:7b:74;
    fixed-address 172.16.1.210;
    filename "pxelinux.0";
}
}
```

2、tftp 配置文件 cat /etc/xinetd.d/tftp 中的蓝色部分是需要注意的，实际上 tftp 是一个 xinetd 的超级守护进程。

```
# cat /etc/xinetd.d/tftp
service tftp
{
    socket_type      = dgram
    protocol         = udp
    wait             = yes
    user             = root
    server           = /usr/sbin/in.tftpd
    server_args      = -s /tftpboot
    disable          = no
    per_source       = 11
    cps              = 100 2
    flags            = IPv4
}
```

启动 tftp 与 dhcp 服务

```
/etc/init.d/xinetd restart  #启动 tftp 服务
/etc/init.d/dhcpd restart  #启动 dhcp 服务
```

下面要实现的是针对安装目录树的网络发布，简单的是使用 nfs 配置文件只有一行就可以了，在 /etc/exports 文件中增加 `/InstallOs *(ro)` 这么一行就可以了。但为了这个安装目录树的通用性，要同时将 InstallOs 文件夹作为 httpd 和 ftp 站点发布出去，方法是在默认安装的 apache 和 vsftp 的根目录中建立一个软连接。

```
ln -s /InstallOs /var/www/html/InstallOS #建立 apache 的软连接
ln -s /InstallOs /var/ftp/InstallOS      #建立 vsftp 的软连接
```

这样就可以提供 nfs/ftp/http 三种网络安装协议的选择。

附上一个安装目录树的扩展价值，使其成为 yum 源

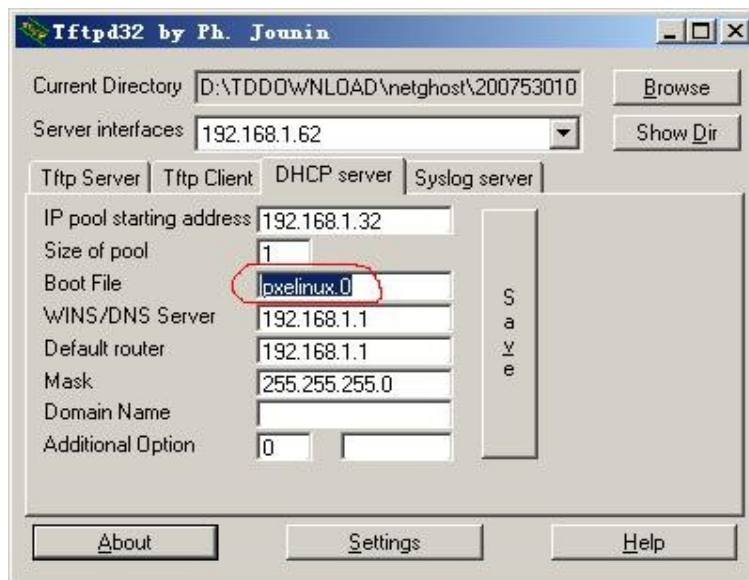
```
cd /InstallOS
rpm -ivh ./Server/createrepo-0.4.11-3.el5.noarch.rpm
createrepo . #创建 yum 仓库
```

在主机/etc/yum.repos.d/目中删除现有的文件建立一个新的 yum 配置文件如下

```
[InstallOs]
name=InstallOs
baseurl=http://安装服务器的 ip 地址/InstallOS/
enabled=1
gpgcheck=0
```

执行 yum clean all 然后就可以使用安装目录树的 yum 源进行软件包的管理了

上述在 Linux 中的实现同样可以在 Windows 主机中通过 3com 的 tftpd32 工具和 Server-u 工具实现，重点是要将 tftboot 和安装目录树在 Windows 主机中构建一份，可以将上面已经构建的文件夹复制到 Windows 主机中。参见下图中的设置部分要将 pxelinux.0 填写到 Boot File 中。将 Server-U 建立的 ftp 通过 Kickstart 整合起来，或者通过 linux askmethod 安装参数选择。



图：tftpd32 设置

ok，现在就可以进行 pxe 的网络引导安装，前提要设置主机使用网卡的 pxe 进行引导。不过现在的 pxe 安装服务器还不是很智能，与使用光盘手动安装一步一步的操作是一样的，创新之处就是不用更换光盘。

Kickstart 的功能，其主要目的是为了减少安装过程中的人机交互，提高安装效率。使用这种方法，只需事先定义好一个配置文件（通常存放在安装服务器上），并让安装程序知道该配置文件的

位置，在安装过程中安装程序就可以自己从该文件中读取安装配置，这样就避免了繁琐的人机交互，实现无人值守的自动化安装。

在 Redhat&Centos 的系统中提供了 system-config-kickstart 工具进行脚本文件的配置，如果能够有一个已经设置好的网络、磁盘分区、安装那些软件包的 Kickstart 脚本文件该多好。实际上在 root 目录下有一个 anaconda-ks.cfg 文件就是当前系统安装的 Kickstart 文件。一般用户可以通过修改这个文件，实现多个相同配置的主机脚本化安装。对于一个 Kickstart 文件分为命令部分、软件包选择部分 %packages、脚本部分（%pre 预安装脚本 和 %post 安装后脚本），对于 Kickstart 文件，命令部分的语言、安装方式、设备规格、键盘设置、引导装载程序配置这些是不能缺少的。anaconda-ks.cfg 中蓝色不会是磁盘分区信息，此文件显示的是从 CD 介质进行安装，并在安装过程中设置了网络参数信息，并可以看到安装后的软件包组信息，具体是那装了那些 rpm 文件需要看 root 帐户的根文件夹下的 install.log 文件。

```
cat anaconda-ks.cfg
# Kickstart file automatically generated by anaconda.

# 以下是 kickstart 文件的命令部分主要设置安装方式、语言、键盘、引导装载程序配置等不可缺少的信息
install
cdrom
lang en_US.UTF-8
keyboard us
xconfig --startxonboot
network --device eth0 --bootproto static --ip 192.168.1.59 --netmask 255.255.255.0
--gateway 192.168.1.1 --nameserver 221.11.12.14 --hostname tes
rootpw --iscrypted $1$aKhtFLuj$gn/LxPNxVXUekzLcgAnJz1
firewall --enabled --port=22:tcp
authconfig --enablesshadow --enablemd5
selinux --enforcing
timezone --utc Asia/Shanghai
bootloader --location=mbr --driveorder=sda --append="rhgb quiet"
# The following is the partition information you requested
# Note that any partitions you deleted are not expressed
# here so unless you clear all partitions first, this is
# not guaranteed to work
# clearpart --linux --drives=sda
# part /boot --fstype ext3 --size=100 --ondisk=sda
```



```
#part pv.2 --size=0 --grow --ondisk=sda
#volgroup VolGroup00 --pesize=32768 pv.2
#logvol swap --fstype swap --name=LogVol01 --vgname=VolGroup00 --size=512 --grow --
maxsize=1024
#logvol / --fstype ext3 --name=LogVol00 --vgname=VolGroup00 --size=1024 --grow

%packages # 此部分开始是系统安装的软件包的选择部分
@editors
@text-internet
@gnome-desktop
@dialup
@core
@base
@base-x
@graphics
@printing
@graphical-internet
device-mapper-multipath
xorg-x11-server-Xnest
libsane-hpaio
-sysreport
```

关于 Kickstart 配置的具体选项可以参考 Redhat 提供的 Installation_Guide 文档，其中有详细的描述和介绍。

文档 Redhat 网站下载地址如下：

http://www.redhat.com/docs/manuals/enterprise/RHEL-5-manual/zh-CN/pdfs/Installation_Guide.pdf

文档 Centos 网站下载地址如下：

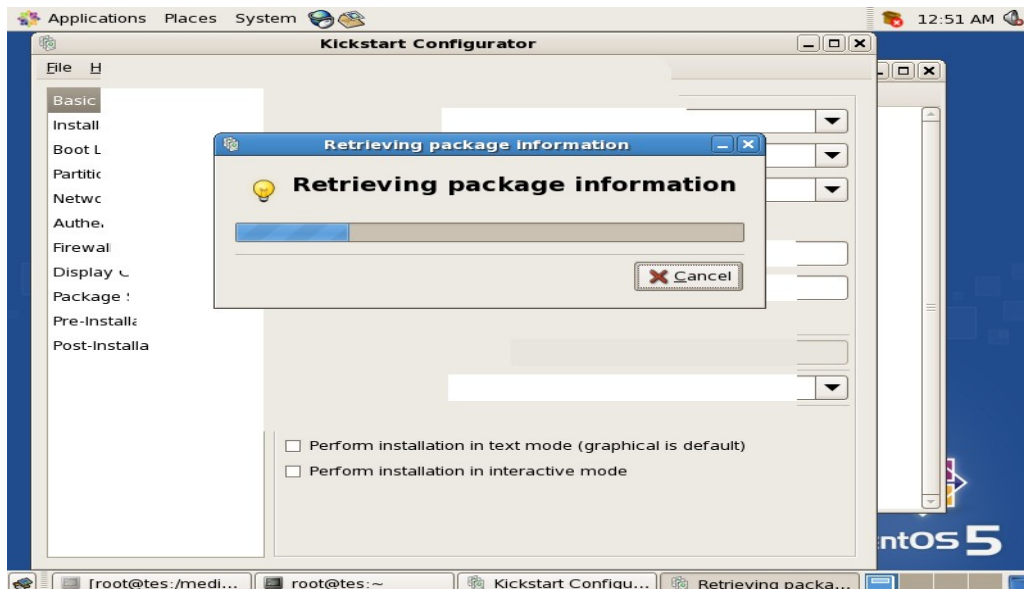
http://www.centos.org/docs/5/html/5.2/pdf/Installation_Guide.pdf

上述事例中的 anaconda-ks.cfg 文件中没有使用 %post 部分脚本，故此在安装完成的系统上出 IP 地址信息设置完成，任何服务器上的配置都需要一步一步的操作完成，这个方式对于一个群集站点的部署来说是很不理想的。通过使用 %post 安装后的脚本，可以通过默认的 shell 脚本语法实现文件的下载、软件包的编译等工作，从而是安装后的主机可以直接投入到使用中，减少准备时间。以下是一个 %post 脚本事例，通过此段代码可以完成在 PXE 安装阶段安装 heartbeat-2.1.4，当然也可以从 ftp 上将 heartbeat-2.1.4 需要的配置文件下载到指定的目录下，以节省配置时间。只要有

了脚本可以完成很多想象的事情！至于能够作什么只有管理员们自己在工作中慢慢体会了。

```
%post
cd /usr/src/
wget ftp://172.16.1.19/libnet.tar.gz
wget ftp://172.16.1.19/heartbeat-2.1.4.tar.gz
tar -xvf libnet.tar.gz
cd libnet
./configure;make;make install;cd ..
tar -xvf heartbeat-2.1.4.tar.gz
cd heartbeat-2.1.4
./ConfigureMe configure;make;make install;
```

安装过程究竟要安装那些软件包合适，不能完全通过已安装系统的 anaconda-ks.cfg 文件获得，Redhat&Centos 提供的 system-config-kickstart 配置工具提供了一个便捷的选择方式，不过限于 Redhat 目前的商业的授权问题，可以通过 Centos 对应的版本进行软件包的选择，具体如下图所示：



图：system-config-kickstart 加载



图：system-config-kickstart 配置软件包

以下是一个 RHEL5.3&Centos5.2 的可用的软件包组的清单，可以工具自己的需要选择。其中蓝色部分为当前版本能够提供的全部的软件包组名称。

```
#platform=x86, AMD64, or Intel EM64T
# System authorization information
auth --useshadow --enablemd5
# System bootloader configuration
bootloader --location=mbr
# Partition clearing information
clearpart --none
# Use graphical install
graphical
# Firewall configuration
firewall --enabled
# Run the Setup Agent on first boot
firstboot --disable
# System keyboard
keyboard us
# System language
```



```
lang en_US
# Installation logging level
logging --level=info
# Use CDROM installation media
cdrom
# SELinux configuration
selinux --enforcing
# System timezone
timezone America/New_York
# Install OS instead of upgrade
install
# X Window System configuration information
xconfig --defaultdesktop=GNOME --depth=8 --resolution=640x480

%packages
@server-cfg
@base-x
@gnome-desktop
@ftp-server
@legacy-software-development
@development-libs
@base
@x-software-development
@sound-and-video
@mail-server
@network-server
@gnome-software-development
@admin-tools
@kde-software-development
@printing
@web-server
```

@mysql
@openfabrics-enterprise-distribution
@development-tools
@chinese-support
@java-development
@legacy-network-server
@system-tools
@kde-desktop
@legacy-software-support
@sql-server
@virtualization
@graphical-internet
@cluster-storage
@text-internet
@ruby
@office
@games
@dialup
@smb-server
@authoring-and-publishing
@graphics
@clustering
@emacs
@java
@dns-server
@engineering-and-scientific
@editors
@news-server

当然这个 system-config-kickstart 配置工具并不局限于软件包的选择，可以设置一个 Kickstart 文件的命令、软件包选择、脚本等全部的信息。

结合 anaconda-ks.cfg 安装配置文件和 system-config-kickstart 配置工具可以定制出特定服

务器的安装 ks 文件，不过不同功能的服务器众多的 ks 配置文件该如何对用起来呢？是特定的主机可以识别相应的 ks 配置文件而不出现错误呢？这个就要说说位于 tftpboot 文件夹下的 pxelinux.cfg 的作用，这个 pxelinux.cfg 文件夹就是根据主机的 MAC 地址对用的配置文件名称识别具体的主机，从而实现特定主机使用相应的 ks 配置文件完成 PXE 的安装及系统的配置。

默认 PXE 配置写入 /tftpboot/pxelinux.cfg/default 中。当计算机在 PXE 期间接收到 DHCP 地址时，就用特定顺序搜索 /tftpboot/pxelinux.cfg 下的配置文件，第一个找到的文件被用作请求计算机的启动配置。搜索顺序的确定方法是：优先搜索以客户端机器的网卡地址命名配置文件（以 '01' 开头，用 '-'（dash）分隔，用小写字母表示 MAC 地址），并把请求 DHCP 地址转换成 8 个 16 进制数字，通过扩展子网对配置目录搜索第一个匹配的文件名（每一次搜索后从右到左删除一位数字），最后使用 default 这个默认的配置。这个是标准的解释不过是不是很麻烦还要进行一个进制转换的操作，不过计算机底层只是别这个没有办法的呀。

例如，网卡的 MAC 地址是 88:99:AA:BB:CC:DD 而 IP 地址是 192.0.2.91，客户会按照如下顺序尝试配置文件名，其中 IP 地址为 192.0.2.91，转换为十六进制数是 C000025B

```
/tftpboot/pxelinux.cfg/01-88-99-aa-bb-cc-dd
/tftpboot/pxelinux.cfg/C000025B
/tftpboot/pxelinux.cfg/C000025
/tftpboot/pxelinux.cfg/C00002
/tftpboot/pxelinux.cfg/C0000
/tftpboot/pxelinux.cfg/C000
/tftpboot/pxelinux.cfg/C00
/tftpboot/pxelinux.cfg/C0
/tftpboot/pxelinux.cfg/C
/tftpboot/pxelinux.cfg/default
```

所以为了优先使每台安装的主机都匹配上 ks 文件就需要获取主机的 MAC 地址并定义好网卡地址命名配置文件。

网卡的 MAC 地址是 88:99:AA:BB:CC:DD 的网卡地址命名配置文件 01-88-99-aa-bb-cc-dd

```
# cat 01-88-99-aa-bb-cc-dd
default linux
label linux
    kernel vmlinuz
    append ip=192.168.1.214 netmask=255.255.255.0 gateway=192.168.1.100
    ks=http://192.168.1.100:/InstallOs/webserver1.cfg initrd=initrd.img
```

其中红色部分的作用对于一些较新型号的兼容网卡是很有意义的可以避免无法获取 IP 地址而终止安装过程的问题。

这样的主机安装和配置过程只需要一台 PXElinux 安装服务器就可以解决问题，安装之前需要获知的即使主机的 MAC 地址，这个应该很简单，商业的服务器的说明书中都是带有网卡 MAC 地址说明的，当然也是可以通过其他途径获得。而且这个 PXE 安装服务器可以便携使用到笔记本电脑的 windows 环境中，之需要将配置好的 ks 配置文件放置在使用 Serve-U 的 ftp 的目录下就可以修改 pxelinux.cfg 中的文件配置完成定制操作。

本文作为系列独辟蹊径网络安装的系列文章的第三篇，希望对 Linux 用户能够有所提示和启发，欢迎就相关问题进行交流。

作者简介: CU 网友 kns1024wh，目前从事 Linux 群集方面的具体工作，之前做过多年的 IT 技术支持、MCT 讲师、及 REDFLAG 的技术合作，技术专长群集、unix 主机、AD 部署等，您可以通过电子邮件 lvsheat@qq.com 或者 Chinaunix 社区与他取得联系。

二探 Cherokee: HTTPS 主机及 PHP 支持设置

ChinxUnix 网友: eScaPedd

在第一期的 Cherokee 初探中，我们大概了解了 Cherokee 这个，web 服务器新贵的一些基础知识，今天我们开始进一步的学习。

通过 Cherokee 的 web 管理端，你可以很方便的设置基于 openssl 的 https 站点。有一个问题需要先说明一下，现在最新的 Cherokee 版本是 0.11.6，在 0.11.5 版本之前，Cherokee 在加载 ca 证书的时候有一个 bug，具体表现就是在启动 https 主机的时候，你会看到类似这样的错误：

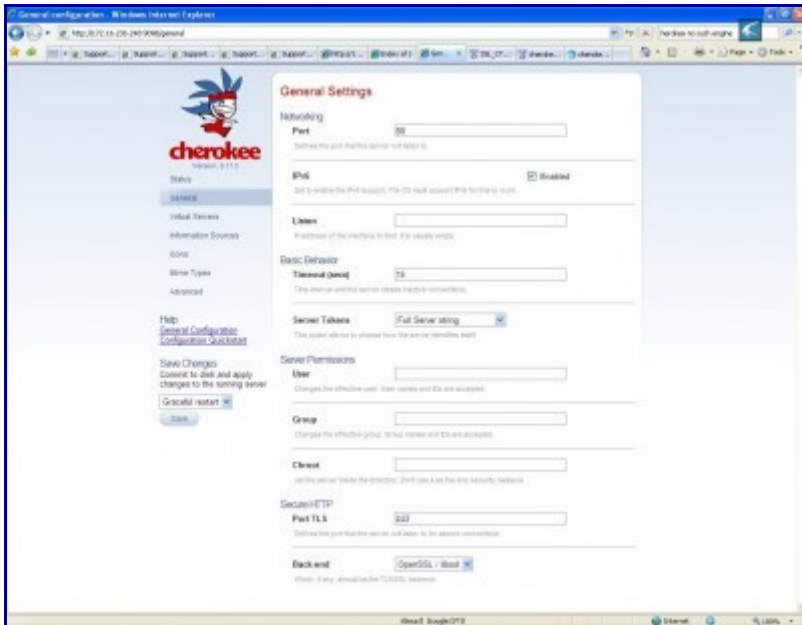
```
cryptor_libssl.c:204: ERROR: OpenSSL: Can not use certificate file '/etc/ssl/cherokee.crt' :  
error:2606A074:engine routines:ENGINE_by_id:no such engine
```

假设你的 Cherokee 安装在 /usr/local/cherokee，我们先来建立自签名的 https 证书，以 root 用户在命令行下依次输入如下命令：

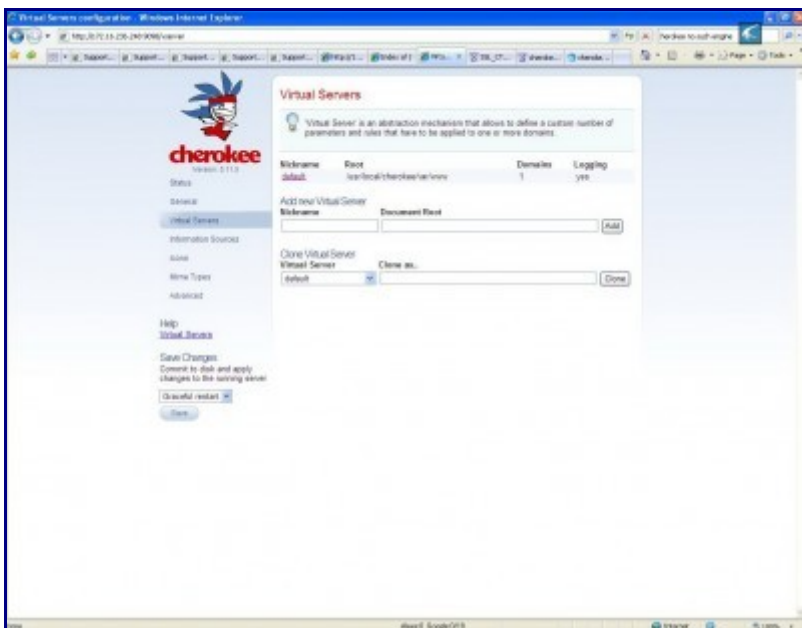
```
1 cd /usr/local/cherokee/etc/cherokee/ssl  
2 openssl req -new -x509 -nodes -out server.crt -keyout server.key  
3 cat server.crt server.key > server.pem
```

这里面进行第二步的时候需要回答一系列问题，比如你的国家代码、省份、城市、证书用在的域名等等。

下面打开 Cherokee 的 web 管理端，选择左侧的“General”，在右侧最下方的“Secure HTTP”里面分别设置 https 的端口，以及后端的类库（也就是 openssl/libssl），如下图：



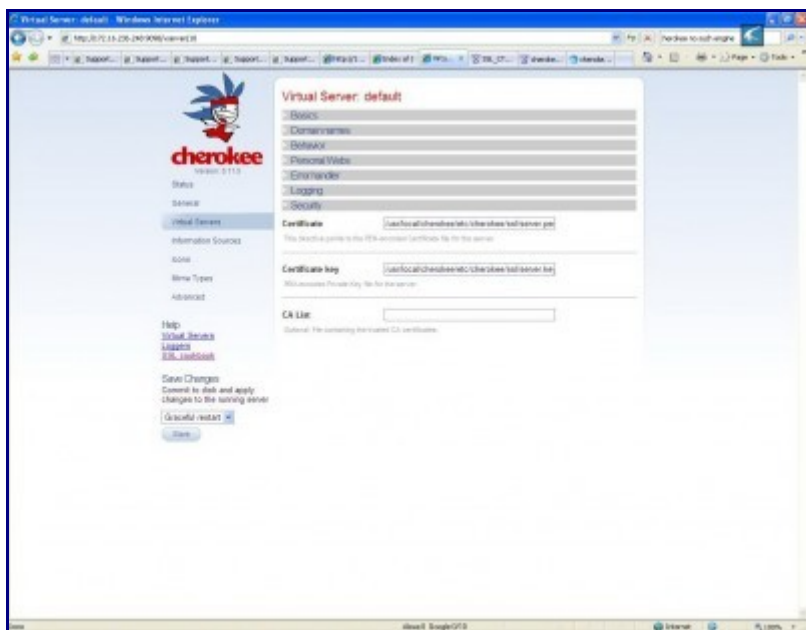
然后选择左侧导航的“Virtual Servers”，这里要注意了，cherokee 的 https 虚拟主机只能设置在默认的 default 主机内，以保证一个 ip 地址只有一个 https 主机：



进入虚拟主机详细设置，点击栏目“Security”，输入证书的路径，对于刚才生成的证书来说，证书和证书 key 文件的路径分别是：

- 1 /usr/local/cherokee/etc/cherokee/ssl/server.pem
- 2 /usr/local/cherokee/etc/cherokee/ssl/server.key

填写好以后，如下图：



好了，基本的设置就这么简单，下面启动虚拟主机，或者重启一下 cherokee 的 worker 进程即可。可以点击左侧导航的“Save”按钮来完成 Graceful restart，也可以回到“Status”菜单启动 cherokee。

最后，打开浏览器，输入 https://172.16.236.248（这个 ip 是我的 cherokee 的 ip）



cherokee 下设置 local 与 remote 方式的 php fastcgi

还是先假设一下服务器环境：

cherokee 的 ip 地址是：172.16.236.191

cherokee 安装在 /usr/local/cherokee

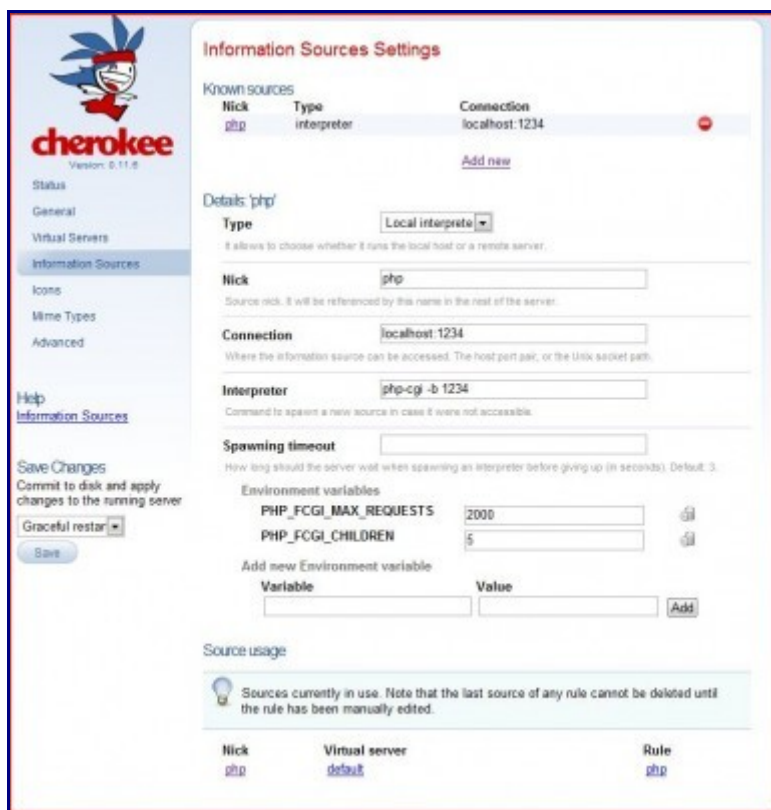
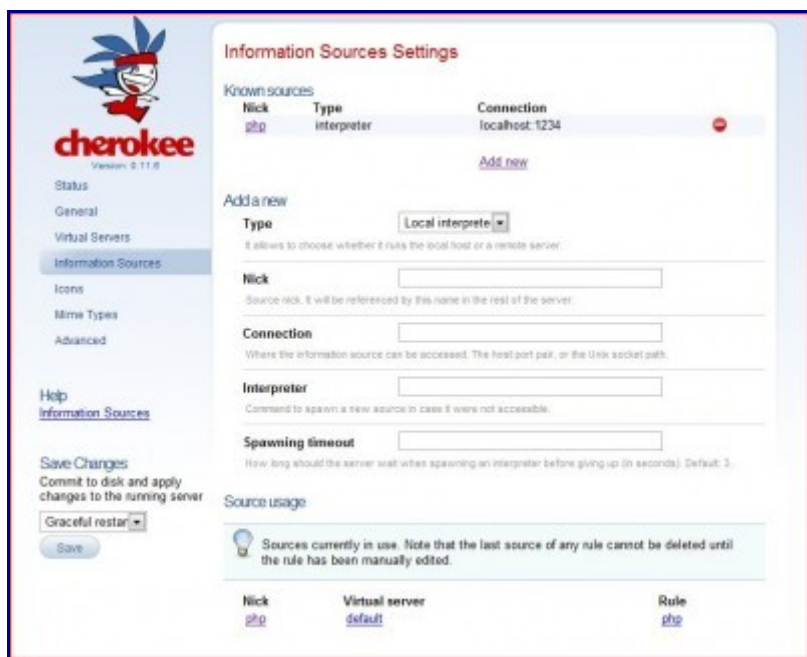
php 安装在 /usr/local/php5，编译了 fastcgi 支持

与 zeus、lighttpd 一样，cherokee 支持将 php 作为本地的 fastcgi 解释器运行，也支持将 php 作为 Remote 方式运行。

默认情况下，cherokee 选择 Local interpreter 作为 php 的运行方式，并且已经附送了一个默认的 php 配置，如果你的 php 可执行文件位于 PATH 环境变量内，你甚至都不用做任何配置就可以直接使用。

下面我们来看看第一种情况如何进行配置：

点击左侧导航栏的 “Information Sources”，在右侧 “Known sources” 下面列出了目前已知的信息源，点击默认的 “php”，进入 php 设置：



其中的各项参数的解释：

Type: “Local Interpreter”（就是本地解释器），一般用在 fastcgi 的可执行程序与 cherokee 运行在同一台服务器的情况。“Remote Host”（就是远程主机），一般用在 fastcgi 的可执行程序与 cherokee 不在同一服务器的情况，或者为了保持 cherokee 和 fastcgi 进程更好的独立性。

Nick: 这个“信息源”的名称，具体就随意了^^

Connection: “信息源”提供的连接方式，可以设置为“IP 地址:端口”的方式（比如：127.0.0.1:1234），也可以设置为 Unix 的 socket（比如/tmp/php.sock）。

Interpreter: “信息源”的解释器，也就是 fastcgi 的可执行文件以及其运行参数。这里我们将原有值改为：/usr/local/php5/bin/php-cgi -b 1234。

Spawn timeout: 超时设置（单位：秒，默认是 3 秒），用于 cherokee 与 fastcgi 进程通信时的超时阈值，这里采用系统默认值，不用填写即可。

作为本地解释器运行的 fastcgi 进程，cherokee 还支持为其设置特定的环境变量。可以通过下面的“Variable”（变量名）、“Value”（数值）来进行设置。默认情况下，cherokee 为 php 设置两个环境变量：

```
1 PHP_FCGI_MAX_REQUESTS 2000
2 PHP_FCGI_CHILDREN 5
```

这两个变量的意思分别是“每个 php 进程在退出前能够相应的请求数目”、“php 的进程数目”。这两个变量需要根据你的网站的负载量进行合理设置，默认情况下，这样就够用了。

下面来看看如何设置 Remote Host 方式的 fastcgi，还是以 php 为例。

在进行设置之前，我们需要先将 php 的 fastcgi 进程作为一个监听特定 ip 以及端口的守护进程运行起来，你可以借助 cherokee 的 spawn-fcgi（在/usr/local/cherokee/bin目录下）或者打过 fpm 补丁的 php，或者直接使用 php-cgi。下面我们以 root 用户 来使用 cherokee 的 spawn-fcgi 来启动 php：

```
1 /usr/local/cherokee/bin/spawn-fcgi -f /usr/local/php5/bin/php-cgi -a 127.0.0.1 -p 9000 -C 5 -P /tmp/
```

简单解释一下 cherokee 的 spawn-fcgi 的几个参数：

- f fastcgi 可执行文件的位置（绝对路径）
- a fastcgi 进程监听的 ip
- p fastcgi 进程监听的端口
- C 子进程数目（该参数为 php 专用）
- P fastcgi 进程的 pid 文件位置
- u fastcgi 进程所属的 unix 用户
- g fastcgi 进程所属的 unix 用户组

为了确定 remote 方式的 php 运行成功，我们可以在命令行下输入 telnet 127.0.0.1 9000 来确认一下。然后我们还是进入 cherokee 的“Information Sources”设置，添加一个新的信息源：

Type: 选择“Remote Host”

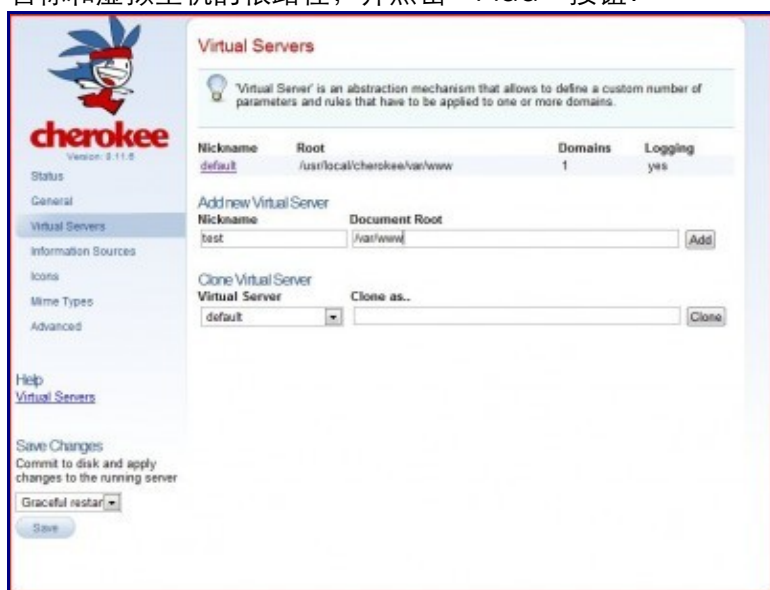
Nick: PHP-Remote

Connection: 127.0.0.1:9000

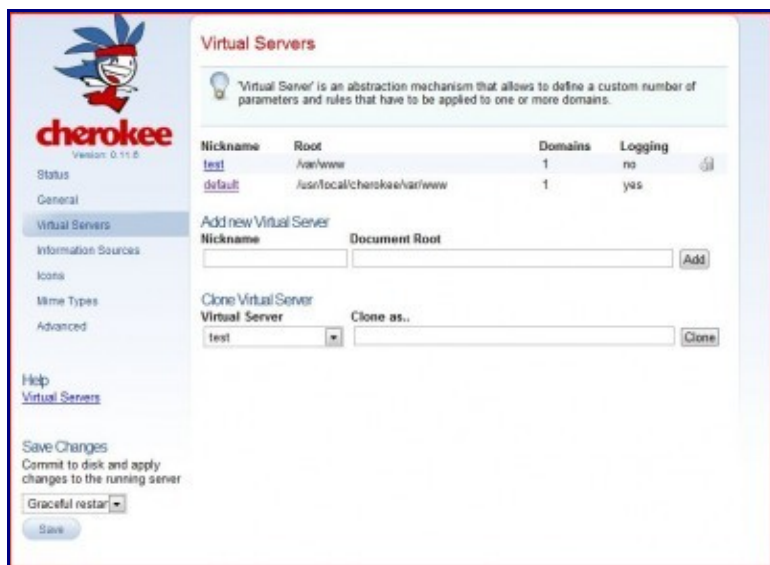
可以看到在 cherokee 设置中已经加入了这个以 remote 方式运行的 php fastcgi。

设置好了信息源，网站怎么知道是用 local 方式的还是 remote 方式的 php 呢？不急，我们还需要进入 cherokee 的虚拟主机设置，选择具体的 fastcgi 解释器。其实 cherokee 已经为默认的虚拟主机配置好了 php（local 方式），如果你也在之前正确配置好了 php-cgi 的路径，那么你可以直接使用默认的虚拟主机配置来运行 php 程序了。如果你希望在虚拟主机中使用 remote 方式的 php，那么我们就一起来看看吧。为了明确整个虚拟主机中 remote 方式的 php 配置，我们将新建一个虚拟主机，从头开始 php 的配置。

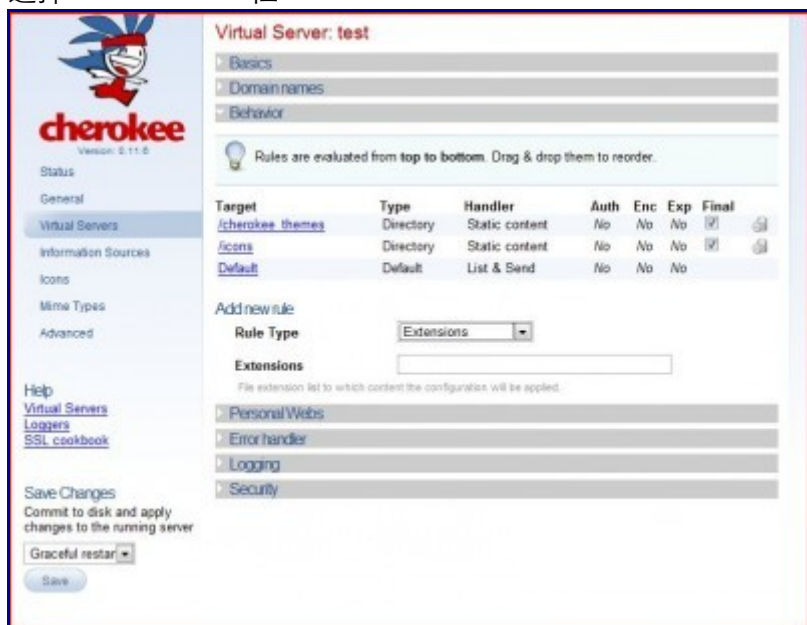
点击左侧导航栏的“Virtual Servers”，在“Add new Virtual Server”下输入新的虚拟主机的名称和虚拟主机的根路径，并点击“Add”按钮：



点击虚拟主机名称，进入虚拟主机详细设置



选择“Behavior”栏：

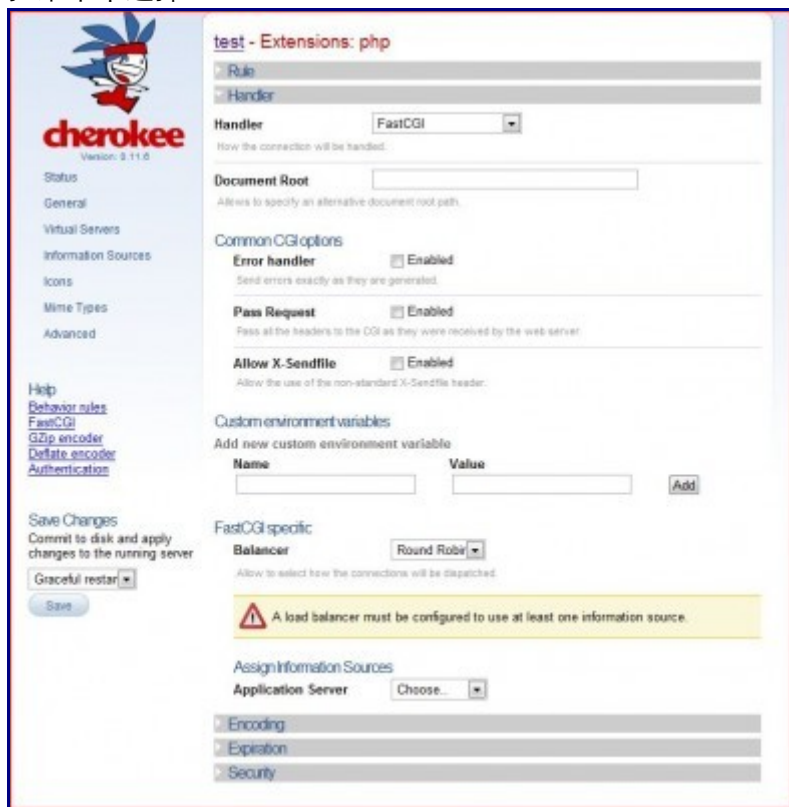


在“Add new rule”下新增一个规则：

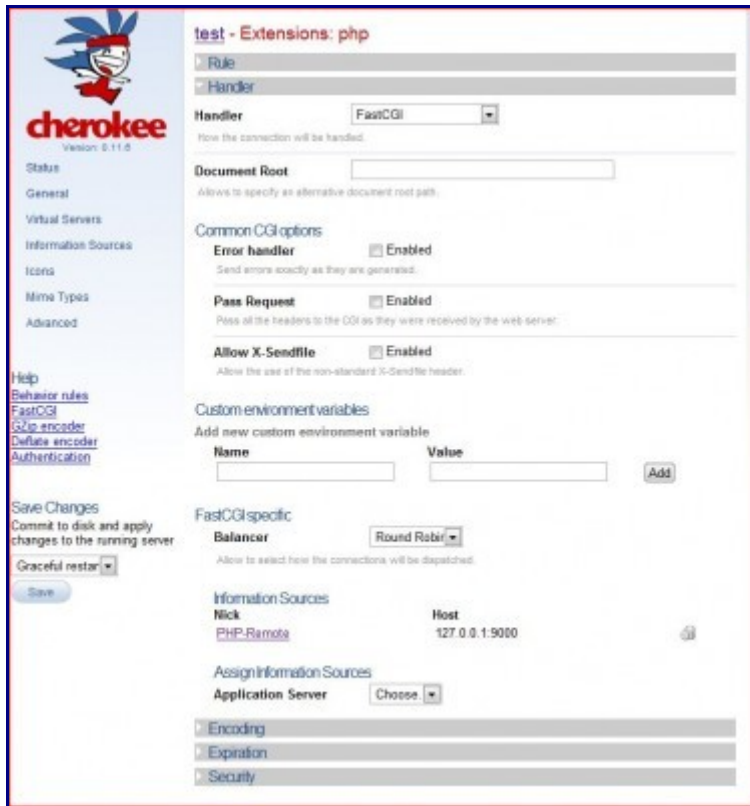
Rule Type: Extensions （新建一个“Extensions”的规则）

Extensions: php （这个规则将作用于所有以 php 为扩展名的文件）

然后 cherokee 会自动进入规则的设置页面，接着我们选择“Handler”栏，在 Handler 的下拉菜单中选择“FastCGI”



在最下方的“Assign Information Sources”中，你可以选择对扩展名为 php 的文件的处理程序（也就是之前的信息源）。经过刚才的设置，这里面我们分别由 php 和 php-remote 两个信息源，选择 PHP-Remote。



至于规则中的其他设置，如果你有兴趣可以自己去研究一下，对于我们之前的目的来说，其他的都使用默认设置就好了。

为了测试的方便，我们先在该虚拟主机根目录下建立一个 php 测试文件：info.php，该文件只有一行代码：

```
1 <?php phpinfo();?>
```

好了，复杂的设置（其实也不复杂）已经做完了，下面来看看效果吧。

先点击左侧导航的“Save Changes”保存虚拟主机设置，然后打开你的浏览器，输入虚拟主机的域名，以及文件名（我这里是：http://172.16.236.191/info.php），如果能看到一系列 php 的系统信息，那么恭喜你，你不要再去找那里配置错误了。

Linux 操作系统下各种文件系统性能比拼

ChinaUnix 网友：Hutuworm

Linux kernel 自 2.6.28 开始正式支持新的文件系统 Ext4。Ext4 是 Ext3 的改进版，修改了 Ext3 中部分重要的数据结构，而不仅仅像 Ext3 对 Ext2 那样，只是增加了一个日志功能而已。Ext4 可以提供更佳的性能和可靠性，还有更为丰富的功能：

1. 与 Ext3 兼容。执行若干条命令，就能从 Ext3 在线迁移到 Ext4，而无须重新格式化磁盘或重新安

装系统。原有 Ext3 数据结构照样保留，Ext4 作用于新数据，当然，整个文件系统因此也就获得了 Ext4 所支持的更大容量。

2. 更大的文件系统和更大的文件。较之 Ext3 目前所支持的最大 16TB 文件系统和最大 2TB 文件，Ext4 分别支持 1EB (1,048,576TB, 1EB=1024PB, 1PB=1024TB) 的文件系统，以及 16TB 的文件。

3. 无限数量的子目录。Ext3 目前只支持 32,000 个子目录，而 Ext4 支持无限数量的子目录。

4. Extents。Ext3 采用间接块映射，当操作大文件时，效率极其低下。比如一个 100MB 大小的文件，在 Ext3 中要建立 25,600 个数据块（每个数据块大小为 4KB）的映射表。而 Ext4 引入了现代文件系统中流行的 extents 概念，每个 extent 为一组连续的数据块，上述文件则表示为“该文件数据保存在接下来的 25,600 个数据块中”，提高了不少效率。

5. 多块分配。当写入数据到 Ext3 文件系统中时，Ext3 的数据块分配器每次只能分配一个 4KB 的块，写一个 100MB 文件就要调用 25,600 次数据块分配器，而 Ext4 的多块分配器“multiblock allocator”（mballoc）支持一次调用分配多个数据块。

6. 延迟分配。Ext3 的数据块分配策略是尽快分配，而 Ext4 和其它现代文件操作系统的策略是尽可能地延迟分配，直到文件在 cache 中写完才开始分配数据块并写入磁盘，这样就能优化整个文件的数据块分配，与前两种特性搭配起来可以显著提升性能。

7. 快速 fsck。以前执行 fsck 第一步就会很慢，因为它要检查所有的 inode，现在 Ext4 给每个组的 inode 表中都添加了一份未使用 inode 的列表，今后 fsck Ext4 文件系统就可以跳过它们而只去检查那些在用的 inode 了。

8. 日志校验。日志是最常用的部分，也极易导致磁盘硬件故障，而从损坏的日志中恢复数据会导致更多的数据损坏。Ext4 的日志校验功能可以很方便地判断日志数据是否损坏，而且它将 Ext3 的两阶段日志机制合并成一个阶段，在增加安全性的同时提高了性能。

9. “无日志”（No Journaling）模式。日志总归有一些开销，Ext4 允许关闭日志，以便某些有特殊需求的用户可以借此提升性能。

10. 在线碎片整理。尽管延迟分配、多块分配和 extents 能有效减少文件系统碎片，但碎片还是不可避免会产生。Ext4 支持在线碎片整理，并将提供 e4defrag 工具进行个别文件或整个文件系统的碎片整理。

11. inode 相关特性。Ext4 支持更大的 inode，较之 Ext3 默认的 inode 大小 128 字节，Ext4 为了在 inode 中容纳更多的扩展属性（如纳秒时间戳或 inode 版本），默认 inode 大小为 256 字节。Ext4 还支持快速扩展属性（fast extended attributes）和 inode 保留（inodes reservation）。

12. 持久预分配（Persistent preallocation）。P2P 软件为了保证下载文件有足够的空间存放，常常会预先创建一个与所下载文件大小相同的空文件，以免未来的数小时或数天之内磁盘空间不足导致下载失败。Ext4 在文件系统层面实现了持久预分配并提供相应的 API（libc 中的 posix_fallocate()），比应用软件自己实现更有效率。

13. 默认启用 barrier。磁盘上配有内部缓存，以便重新调整批量数据的写操作顺序，优化写入性能，因此文件系统必须在日志数据写入磁盘之后才能写 commit 记录，若 commit 记录写入在先，而日志有可能损坏，那么就会影响数据完整性。Ext4 默认启用 barrier，只有当 barrier 之前的数据全部写入磁盘，才能写 barrier 之后的数据。（可通过“mount -o barrier=0”命令禁用该特性。）

Ext4 随 Linux kernel 2.6.28 正式发布已有数周，一直苦于找不到测试用的磁盘，正巧年前 Intel

送来几块 SSD 测试样品，这两天就顺带把 SSD 也测了。测试所使用的 Linux 内核版本为 2.6.28.2，测试工具为 IOzone 3.318。

[IOzone](#) 测试命令为：

```
time /opt/iozone/bin/iozone -a -s 4G -q 256 -y 4 > | /root/ext4-iozone-stdout.txt
```

上述命令的说明如下：

Auto Mode

File size set to 4194304 KB

Using Maximum Record Size 256 KB

Using Minimum Record Size 4 KB

Command line used: /opt/iozone/bin/iozone -a -s 4G -q 256 -y 4

Output is in Kbytes/sec

Time Resolution = 0.000001 seconds.

Processor cache size set to 1024 Kbytes.

Processor cache line size set to 32 bytes.

File stride size set to 17 * record size.

测试结果除了表明 Intel SSD 的读写速度快得令人咋舌之外，还可以说明 Ext4 的各方面性能都超过了上一代 Ext3，甚至在大多数情况下，比没有日志功能的 Ext2 还要快出不少：

		recl en	writ e	rewr ite	rea d	rere ad	om rea d	om writ e	dkw rea d	recor d rewrit e	strid e rea d	fwrit e	frew rite	frea d	frerea d
Ext2 real 28m12 .718s user 0m10. 725s sys 5m8.2 65s	4	218, 680	216, 596	630, 248	245, 802	88,7 00	138, 065	106, 112	1,882 ,623	73,5 38	214, 175	218, 364	566, 570	247,38 1	
	8	215, 308	218, 690	556, 064	246, 260	154, 680	150, 052	188, 397	2,462 ,367	130, 896	217, 157	216, 647	583, 808	248,39 7	
	16	216, 457	216, 843	575, 046	245, 701	258, 660	158, 750	306, 842	2,654 ,320	220, 939	216, 061	218, 140	598, 174	246,58 1	
	32	217, 925	214, 289	537, 976	243, 081	394, 013	167, 002	464, 240	2,397 ,831	340, 775	217, 434	219, 353	583, 463	246,34 1	
	64	215, 460	219, 256	527, 919	244, 362	503, 227	162, 917	609, 546	2,546 ,079	456, 243	216, 875	217, 692	571, 707	244,26 4	
	128	219, 081	216, 173	540, 831	242, 526	609, 750	161, 442	721, 701	2,656 ,729	551, 122	217, 780	217, 427	579, 271	242,29 1	
	256	216, 091	217, 631	565, 111	245, 157	654, 274	173, 955	870, 547	2,574 ,261	634, 835	216, 638	219, 693	563, 735	247,10 1	

Ext3 real 27m42.449s user 0m11.529s sys 7m17.049s	4	218,242	213,039	482,132	243,986	88,007	156,926	105,557	1,540,739	75,010	216,028	216,432	522,704	243,385
	8	218,390	217,915	544,892	244,979	152,424	190,454	181,486	1,945,603	130,737	218,364	216,431	530,853	243,222
	16	218,083	217,683	561,038	244,506	255,244	200,032	300,212	2,096,495	221,329	216,930	216,661	514,177	244,069
	32	216,258	217,013	569,246	243,811	389,745	198,275	446,462	1,934,853	338,785	216,809	219,296	530,634	243,446
	64	218,850	217,711	577,529	243,725	497,689	201,693	589,535	2,036,412	450,449	219,387	214,900	514,353	244,809
	128	220,234	215,687	530,519	241,615	608,244	199,619	714,295	1,992,168	553,022	217,828	218,454	513,596	241,510
	256	216,011	220,188	592,578	242,548	642,341	199,408	834,240	2,092,959	624,043	217,682	218,165	529,358	242,878
Ext4 real 27m3.485s user 0m10.847s sys 6m9.578s	4	221,823	216,992	532,488	273,668	85,210	183,195	103,036	1,862,817	74,781	225,841	220,620	523,799	272,848
	8	226,028	218,580	561,960	272,036	154,972	216,505	178,482	2,135,372	132,506	227,423	215,766	641,021	271,328
	16	222,241	217,746	547,548	270,895	260,899	223,895	295,288	2,095,966	223,135	226,055	216,210	621,287	273,475
	32	220,121	213,025	240,426	247,628	345,210	175,977	451,631	2,145,351	342,236	225,796	213,427	598,331	269,759
	64	223,983	214,437	308,696	551,577	754,941	225,897	523,130	2,218,016	448,086	227,030	214,706	582,795	272,323
	128	222,576	217,816	624,636	271,293	644,500	224,997	720,468	2,308,315	582,943	225,971	217,373	552,335	274,237
	256	221,202	222,238	541,685	270,898	671,748	228,085	845,494	2,215,381	643,715	225,411	219,166	580,066	273,342

Kbytes
/sec

注:

1. 关于 IOzone 测试方法, 参考 Ben Martin 的文章: [IOzone for filesystem performance benchmarking](#)
2. 关于 Ext4 的相关内容, 参考 Kernel Newbies 专页: <http://kernelnewbies.org/Ext4>

自上面的测试报告发布以来，社会各界纷纷来电来函，给出了“Ext4 我看行”，“Ext4 牛牛牛”，“用了 Ext4，嘿，还真对得起这块盘”，“Ext4，碎片去无踪，性能更出众”等等热情洋溢的评论，兄弟我深受鼓舞。在对 Ext4 颇不低俗的性能进行表扬的同时，人民群众还提出了新的要求，希望看到与现有同类文件系统性能横向比拼的结果。为了满足广大群众的热切需求，今天做了 Ext2、Ext3、Ext4、XFS、JFS、ReiserFS 和 Btrfs 的全面性能测试，对比结果如下：

		recl en	writ e	rewr ite	rea d	rere ad	ran do m rea d	ran do m writ e	bkwd read	recor d rewrit e	strid e read	fwrit e	frew rite	frea d	frerea d
Ext2 real 27m23 .545s user 0m9.1 84s sys 4m40. 925s	4	218, 554	218, 662	668, 245	245, 208	85,2 35	175, 284	101,8 64	1,911 ,579	74,3 68	215, 014	217, 680	583, 169	244,83 1	
	8	216, 064	220, 489	618, 661	243, 444	146, 695	215, 002	176,9 08	2,314 ,305	129, 333	215, 772	218, 571	605, 436	241,14 4	
	16	218, 874	219, 173	609, 563	242, 592	246, 442	220, 306	292,1 14	2,438 ,520	218, 357	216, 497	219, 774	596, 211	244,65 2	
	32	219, 584	217, 859	611, 712	244, 445	372, 023	224, 151	442,6 29	2,423 ,574	334, 999	218, 829	219, 553	606, 186	244,34 6	
	64	216, 320	219, 400	622, 218	243, 761	478, 747	227, 018	580,9 60	2,633 ,117	439, 129	215, 956	220, 166	608, 210	243,95 0	
	128	219, 286	218, 318	614, 559	243, 862	579, 724	224, 346	690,9 45	2,467 ,787	537, 440	219, 420	218, 330	607, 326	244,33 7	
	256	216, 027	220, 509	635, 633	245, 385	614, 120	227, 099	842,2 20	2,567 ,169	614, 753	217, 744	219, 897	617, 728	243,78 0	

Ext3 real 27m26. .675s user 0m9.3 23s sys 6m43. 199s	4	215,478	218,294	604,858	246,148	86,758	151,858	102,113	1,618,769	74,979	214,446	218,569	568,659	242,772
	8	217,695	218,777	609,377	244,520	148,608	190,313	171,903	1,830,365	128,957	218,752	217,501	603,465	246,861
	16	215,683	219,822	643,396	248,815	252,255	202,374	290,681	1,915,488	219,805	216,519	219,033	568,086	243,985
	32	219,341	217,364	603,462	245,273	379,512	208,740	435,756	2,069,791	335,303	218,578	219,086	604,572	245,578
	64	216,192	220,186	665,646	244,234	481,101	212,185	567,750	2,106,710	442,926	217,075	220,042	618,134	244,870
	128	218,503	219,151	616,479	244,711	589,927	212,355	683,943	2,080,136	541,597	219,271	218,514	600,308	244,782
	256	217,255	219,213	637,417	248,555	636,309	211,793	839,907	2,125,314	618,958	218,424	218,954	600,894	245,523
Ext4 real 26m15. .771s user 0m9.6 09s sys 5m36. 096s	4	225,586	224,616	532,721	270,174	82,398	180,231	102,935	1,683,476	74,990	224,893	224,186	277,025	584,868
	8	226,896	221,965	274,314	629,603	239,913	219,223	156,964	2,026,184	127,553	226,382	227,019	292,566	593,154
	16	222,929	229,392	272,022	599,360	390,533	223,951	256,831	2,114,002	212,773	227,728	225,332	602,317	269,289
	32	223,703	221,025	566,981	262,980	380,787	227,198	435,774	2,157,610	338,629	226,208	225,816	591,102	268,725
	64	222,236	223,813	593,481	268,002	522,365	227,268	582,830	2,216,225	457,252	228,294	223,269	614,376	269,657
	128	221,681	222,859	285,304	623,990	915,430	231,380	663,383	2,276,039	573,624	226,616	224,477	583,965	271,159
	256	220,129	221,608	279,101	657,815	889,963	232,256	770,220	2,212,083	610,074	225,877	224,642	288,852	603,933

XFS	4	217,459	214,348	244,600	239,647	78,424	161,673	107,643	1,711,976	74,828	218,638	216,903	227,675	240,492
	8	220,514	219,994	241,854	951,125	248,623	207,750	161,175	2,214,681	127,363	218,892	211,613	219,749	240,286
	16	220,776	220,740	226,186	242,746	224,486	219,963	318,503	2,480,826	223,664	221,106	221,694	669,693	262,346
	32	218,481	216,919	255,232	799,715	593,083	225,213	408,816	2,278,034	337,026	222,153	213,330	226,093	241,351
	64	216,921	210,031	228,877	244,161	430,479	225,432	612,872	2,568,460	448,155	224,650	209,745	221,768	239,888
	128	216,341	218,284	232,550	242,283	559,400	227,915	768,251	2,545,859	568,945	222,605	216,060	714,911	258,397
	256	221,444	220,462	469,309	321,275	757,895	225,641	806,251	2,689,152	609,077	219,703	219,815	226,021	240,108
JFS	4	196,819	218,121	455,194	258,062	85,910	169,163	142,814	1,949,282	92,246	212,270	217,272	450,996	258,928
	8	211,794	215,030	445,452	258,082	147,556	208,623	236,197	2,214,516	159,402	213,076	218,146	451,013	258,767
	16	209,028	214,714	455,795	257,989	248,145	218,747	399,935	2,587,176	269,527	214,271	215,626	439,736	258,830
	32	210,454	218,204	448,067	258,568	380,325	220,682	591,781	2,379,845	409,061	213,292	217,683	439,114	259,160
	64	214,862	215,636	450,430	258,413	487,982	222,660	750,139	2,613,574	522,050	216,365	217,830	444,546	259,553
	128	213,152	218,084	449,185	258,775	592,912	148,064	933,583	2,728,229	667,878	219,155	213,461	466,396	262,608
	256	213,119	213,514	477,804	258,508	631,376	129,060	1,055,205	2,727,762	717,814	213,771	217,569	463,931	258,291

ReiserFS real 28m16.075s user 0m10.134s sys 7m38.726s	4	208,422	217,932	190,459	223,684	72,142	175,210	99,820	1,726,654	74,081	211,433	216,255	540,847	239,091
	8	214,840	218,917	610,592	240,376	146,908	210,703	176,295	2,180,046	128,114	215,368	218,950	580,334	240,261
	16	216,886	217,827	651,240	243,800	249,478	219,967	288,457	2,444,958	213,780	217,438	219,510	592,932	239,510
	32	214,217	219,178	645,366	240,227	369,030	224,795	434,556	2,372,823	329,006	217,055	217,110	554,072	240,046
	64	217,160	219,013	623,512	239,875	470,435	224,619	560,790	2,429,552	421,615	218,043	218,734	587,074	239,802
	128	215,163	219,348	624,117	240,364	572,926	226,463	683,686	2,530,721	530,024	215,932	218,157	577,617	240,975
	256	217,058	218,839	643,857	240,141	615,875	224,530	825,364	2,631,201	590,909	217,256	218,942	567,422	240,061
Btrfs real 59m41.974s user 0m10.382s sys 18m40.804s	4	214,203	215,879	265,389	280,244	65,367	10,520	16,032	773,644	34,260	94,105	194,357	269,380	268,254
	8	218,318	215,978	270,100	275,160	109,687	26,015	40,447	1,366,911	89,489	141,760	214,990	273,446	271,531
	16	219,477	215,738	284,876	282,111	178,662	47,881	72,623	1,864,106	150,255	177,508	219,348	264,935	273,140
	32	218,835	220,886	281,550	277,054	246,933	130,288	181,774	2,573,279	250,936	204,491	219,736	281,894	268,989
	64	216,758	218,890	282,026	275,127	312,803	195,963	299,685	3,270,521	354,200	202,362	220,928	281,331	271,676
	128	216,509	220,235	279,353	274,813	368,053	215,445	343,696	3,739,759	414,950	216,029	219,711	276,973	272,569
	256	220,333	216,113	277,920	273,271	405,215	215,495	393,731	4,015,851	474,052	213,849	220,372	283,035	277,594

Kbyte
s/sec

本次测试所使用的 Linux kernel 版本为 2.6.29-rc3，文件系统性能测试工具为 IOzone 3.318。

从测试结果可以看出，Ext4 的综合性能位居现有文件系统之首，JFS、ReiserFS 在读性能方面亦有不俗表现。Btrfs 的小块数据读写性能与平均水平相差甚远，是导致其本次测试总时间超出平均时间两倍的主要原因。较之其它成熟的文件系统，Btrfs 投入生产系统运作可能尚需时日。

至于一小撮不明真相别有用心的人群提出还要测 NTFS、Vfat、MSDOS 等“文件系统”，恕时间有限，不予受理。

OpenBSD. Nginx. MySQL. PHP 环境搭建手册

ChinaUnix 网友: young_king

所谓的 O.N.M.P.是指 OpenBSD、Nginx、MySQL、PHP(fastcgi)，是利用 OpenBSD 及其软件包搭建的时下流行的 MySQL+PHP 应用环境，下面对主要的软件做下介绍：

OpenBSD：可能是这个星球最安全的操作系统了，在十几年的时间里面只被发现了两个远程安全漏洞；

Nginx：来自俄罗斯的 HTTP 软件，据说性能达到老牌的 Apache 十倍！并且极其节约资源，是单台服务器跑 PHP 应用的首选。官方文档声称能达到 5W 个并发连接，生产环境下单台双核 2.33G 服务器可以跑到 3W 个并发连接（仅运行 Nginx+PHP-fastcgi）；

MySQL：老牌的开源数据库软件；

PHP：这个几乎无人不知，和 MySQL 一起构成了开源环境下最强的建站组合。

下面以 [OpenBSD 4.4](#) 为例进行讲解。按照惯例，root 环境：

一、软件环境的搭建

OpenBSD 的安装就不罗嗦了，重点是分区的部分，推荐/var/mysql、/var/log、/var/mail、/var/nginx 这几个目录单独分区，并给予足够的空间。系统安装的最后询问是否默认启动 Ssh 服务时回答“n”。

安装必须的软件包：

```
export PKG_PATH=ftp://ftp.openbsd.org/pub/OpenBSD/4.4/packages/i386/  
pkg_add wget mysql-server php5-fastcgi php5-gd-5.2.6-no_x11 php5-mysql phpMyAdmin  
lighttpd-1.4.19p3 nginx pecl-APC
```

装完后按提示做连接并创建 PHP 工作目录：

```
ln -s /var/www/conf/modules.sample/php5.conf /var/www/conf/modules  
ln -fs /var/www/conf/php5.sample/apc.ini /var/www/conf/php5/apc.ini  
ln -fs /var/www/conf/php5.sample/gd.ini /var/www/conf/php5/gd.ini  
ln -fs /var/www/conf/php5.sample/mbstring.ini /var/www/conf/php5/mbstring.ini  
ln -fs /var/www/conf/php5.sample/mcrypt.ini /var/www/conf/php5/mcrypt.ini  
ln -fs /var/www/conf/php5.sample/mysql.ini /var/www/conf/php5/mysql.ini  
mkdir /var/www/tmp  
chmod 0777 /var/www/tmp  
mkdir /var/nginx/html/pma/  
cp -rf /var/www/phpMyAdmin/* /var/nginx/html/pma/  
vi /etc/login.conf
```

跳到最后，加入 MySQL 所需的修改：

```
mysql:\
:openfiles-cur=2048:\
:openfiles-max=4096:\
:tc=daemon:
```

使修改生效:

```
cap_mkdb /etc/login.conf
```

安装数据库:

```
/usr/local/bin/mysql_install_db
```

二、修改各软件的配置文件

修改 nginx 的默认配置文件:

```
vi /etc/nginx/nginx.conf
```

增大 nginx 的并发连接数:

```
worker_connections 51200;
```

加入对.php 文件的支持:

```
location / {
    root /var/nginx/html;
    index index.php index.html index.htm;
}
```

(加入 index.php)

将下面这段前面的注释都去掉, 改成:

```
location ~ \.php$ {
    root    html;
    fastcgi_pass 127.0.0.1:9000;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME /var/nginx/html$fastcgi_script_name;
    include fastcgi_params;
}
```

注意上面第五行, 这个部分原来是/scripts, 改成/var/nginx/html。

其他的部分请酌情修改。

修改 MySQL 的配置文件 (或从/usr/local/share/mysql 目录中拷贝合适的文件为/etc/my.cnf)。

关于 MySQL 的配置和优化, 网上有很多这方面的文章, 可以自行 Google, 这里就不多讨论了。

贴下偶的 (总内存 512M, MySQL 使用 256M, 在 my-large.cnf 的基础上修改):

在[mysqld]段增加或修改:

```
skip-innodb
skip-networking
max_connections = 600
skip-bdb
skip-name-resolve
```



```
wait_timeout= 5
```

保存退出。

修改 phpMyAdmin 的配置文件，使之可用：

```
vi /var/nginx/html/pma/config.inc.php
```

将下面这行修改成：

```
$cfg['blowfish_secret'] = 'a'; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */
```

（仅是加入了字母 a 而已）

保存退出。

三、系统启动脚本的修改

修改系统启动脚本，使得 MySQL、PHP(fastcgi)、Nginx 可以在系统启动的时候自动启动，免去手动启动的麻烦。

```
vi /etc/rc.local
```

跳到最后，加入下面的部分：

```
rdate -n 210.72.145.44
```

```
# Start MySQL
```

```
if [ -x /usr/local/bin/mysqld_safe ] ; then
```

```
    echo -n 'Starting MySQL...'
```

```
    su -c mysql root -c '/usr/local/bin/mysqld_safe >/dev/null 2>&1 &'
```

```
    echo "DONE"
```

```
fi
```

```
# Start php-fastcgi
```

```
if [ -x /usr/local/bin/spawn-fcgi ] ; then
```

```
    echo -n 'Starting php-fastcgi...'
```

```
    /usr/local/bin/spawn-fcgi -a 127.0.0.1 -p 9000 -C 6 -u www -f /usr/local/bin/php-fastcgi > /
```

```
var/run/fcgi.pid
```

```
    echo "DONE"
```

```
fi
```

```
# Start nginx
```

```
if [ -x /usr/local/sbin/nginx ] ; then
```

```
    echo -n 'Starting nginx...'
```

```
    /usr/local/sbin/nginx
```

```
    echo "DONE"
```

```
fi
```

保存退出。

四、修改默认的 SSH 端口

```
vi /etc/ssh/sshd_config
```

在 #Port 22 的下面新建一行，输入：

Port 49813

使得 Sshd 监听 49813 端口，以避免黑客扫描 22 端口。

vi /etc/rc.conf.local

增加一行，输入：

sshd_flags=""

使得系统启动时启动 Sshd 服务。

保存退出。

reboot

等系统重启完毕，用 top 命令看下，O.N.M.P.环境已经搭建好了！

```
load averages: 0.12, 0.10, 0.08
26 processes: 25 idle, 1 on processor
CPU states: 0.2% user, 0.0% nice, 0.0% system, 0.0% interrupt, 99.8% idle
Memory: Real: 54M/107M act/tot Free: 389M Swap: 0K/502M used/tot
```

PID	USERNAME	PRI	NICE	SIZE	RES	STATE	WAIT	TIME	CPU	COMMAND
4215	_mysql	2	0	276M	33M	sleep	poll	0:00	0.00%	mysqld
16033	www	2	0	21M	6384K	idle	poll	0:00	0.00%	php-fastcgi
6750	www	2	0	21M	1600K	idle	poll	0:00	0.00%	php-fastcgi
20362	www	2	0	21M	1588K	idle	poll	0:00	0.00%	php-fastcgi
14668	www	2	0	21M	1588K	idle	poll	0:00	0.00%	php-fastcgi
21492	www	2	0	21M	1588K	idle	poll	0:00	0.00%	php-fastcgi
19544	www	2	0	21M	1588K	idle	poll	0:00	0.00%	php-fastcgi
28668	www	2	0	21M	1588K	idle	poll	0:00	0.00%	php-fastcgi
9233	_nginx	2	0	11M	11M	idle	kqread	0:00	0.00%	nginx
31257	root	2	0	3348K	2372K	sleep	select	0:00	0.00%	sshd
32730	root	2	0	968K	1452K	sleep	select	0:00	0.00%	sendmail
19445	root	18	0	816K	948K	idle	pause	0:00	0.00%	nginx

写在最后：

终于明白了 OpenBSD 官方力推 packages 的原因！看看吧，步骤多么简单，无须下载软件源码，无须长时间的 make && make install，很短的时间就可以搭建起时兴的 O.N.M.P.环境，而且安全性有很高的保障！

对 MySQL 的管理可以通过 phpMyAdmin 来完成，地址：

<http://xxx.xxx.xxx.xxx/pma/index.php> (xxx.xxx.xxx.xxx 为你服务器的 IP 地址)。默认情况下 MySQL 的 root 用户是没有密码的，请第一时间设置！

Linux 系统下时间处理全过程

ChinaUnix 网友: art_codes

对于一个操作系统，时间能够在开机后正确显示，处理好两点要点就可以了，一是要在关机时候系统存储一个秒数。（按理说存储一个毫秒数应该更精确，但是也许是太占存储空间，反正现在业界形成的规定就是秒数）然后在关机的时候由主板上的电池每隔一秒钟就加一下它的数值就可以。这样系统就能在关机下还能存储住正确时间。在开机的时候在把这个被主板电池操作后的秒数加载回系统来处理就可以。这样一个时钟的存储过程从理论上就可以行得通。

其实本来这样就可以了，但是这个地球毕竟是个球，导致有人是黑天有人是白天，没有办法。所以人们又分了时区，以 GTM 或 UTC 时间为全球时间的基准点，（这两个概念上我希望大家不要

混，虽然起源背景有一些不同，但是到了实际应用中基本上表示的意义都是一样的，那就是全球的基准时间，数值上基本是相同的）为全球的基准时间，和地区的时间。比如中国的 CST 时间（CST = China Standard Time UTC+8:00，大家看到的 UTC+8:00=GTM+8:00）。由于 UTC 和 CST 都是一个数（秒数）所以将那个数存储 bios 的时间里面就是一个很严重的问题。经过测试不同的 UTC 和 CST 不同存储方式直接影响这系统间时间的兼容性。

时间处理方法 1. 存储 UTC 的秒数那么整个系统的开关机过程将为：

关机：CST(时区变化，加减相应的小时)->UTC->bios----在这种情况下，在 bios 中看到的就是 UTC 时间。（由于 bios 中没有时区的概念，所以你要像看懂这个时间代表的是哪里地方的时间，你一定要明白它由谁存的数，还有这个数原本代表了什么意义，如是那个时区的。）

开机:bios->UTC->CST(时区变化)

时间处理方法 2. 存储 CST 的秒数那么整个系统的开关机过程将为：

关机:CST(不用时区变化了，之间化为秒数后)->bios----这个时候你在 bios 中看到的时间就是有时区的。

开机:bios->CST

大家看到这两种方法后应该发现，其实真的不能说那种存储方法更好，其实过程都差不多，但是如果装两个不同操作系统的机器就有麻烦了，怎么说呢，比如一个操作系统(比如 windows)是用 2 方法存储的时间，但是另一个操作系统是用 1 来存储时间的(大家要看到，他是这么存储时间的所以他一定也是用 1 的方法来解析的)，那么在这个后一个操作系统中可能看到正确的时间么？

经过实际测试发现，现在操作系统的默认存储方法都是采用的是第二种方法（就是直接存储 CST 时间，如 UBUNTU windows fedora，不信大家可以自己去测试）。说到这里要说明白，以前看过有兄弟说机器一个是 windows 一个是 linux 双系统的机器里面时间老是错误的情况，那要怎么改呢？windows 咱不懂不说了（主要是找不到在哪里改时间存储方式……），所以只能让 linux 去适应 windows 啦，改 linux 的时间存储方式就是去改/etc/sysconfig/clock 这个文件（redhat 和 fedora 里面有的，ubuntu 默认下就和 windows 是匹配的配置随意大家不用改，而且 ubuntu 好像没有/etc/sysconfig/clock 这个文件……）这个文件的内容如下

```
# cat /etc/sysconfig/clock
# The ZONE parameter is only evaluated by system-config-date.
# The timezone of the system is defined by the contents of /etc/localtime.
ZONE="Asia/Shanghai"
UTC=false
ARC=false
```

看到 UTC 这行，对于他来说 true 就是用 1 的方法方法存储时间，false 就是用 2 的方法来存储时间。为了配合 windows 所以大家只能改成 false 这样就可以万事大吉。

还有一个需要和大家说的就是，大家都说 date 命令改的是系统时间，而 hwclock 命令改的是硬件时间，但是我发现（大家不信可以自己去试试看）用 hwclock --set --date="2009-2-13"根本改变不了系统的 bios 时间（我说的是改后重启系统在 bios 中看到是不是你设定的值），反而用 date -s 可以改 bios 系统时间。这是为什么呢？原来是因为不管你改没改硬件时间，系统在关机的时候都会将当前的系统时间写入 bios（之前的 hwclock 白改……），就像 1 或 2 存储过中的关机过程那样。所以说你用 hwclock --set --date="2009-2-13"确实改了硬件时间，但是在关机的时候，这

个时间又被当前系统时间给覆盖了，有人说 `hwclock --systohc` 可以改变硬件时间，我想说那你仔细想想这个命令有意义么？（你只不过是当前的系统时间和硬件时间同步了一次，只是让关机时候的自动时间覆盖过程看不出来了而已。）

我想说的就是这么多了，由于是刚开始学习 linux，英文也不是很好（其实英语还行，但是一看见一大堆英语在一块就想吐……），所以很多的地方是实践后得出的结果，大家不信可以试试看，如有不对的还请多多指教。谢谢大家！

一些说明：

GTM、UTC 国际的标准基准时间，用来同一时间用的，用户一般用不到，查看命令 `date -u`

CST 一般操作系统都用它为时间，查看命令 `date`

`setup` 命令设定时区超级好用

linux 中有：

系统时间：简单可以理解为加载了时区的时间，如 CST

硬件时间：简单理解为在 bios 中的时间

Ubuntu 8.10 中输入法 ibus 的安装和使用

ChinaUnix 网友：Fanball

Linux 下常见的输入法有 `fcitx` 和 `scim`。`fcitx`，全称“Free Chinese Input Toy for X”，中文名称为“小企鹅输入法”，支持拼音，五笔字型以及区位、二笔。可以输入 UTF-8、GB 18030 编码中的汉字。`fcitx` 小巧好用，兼容性好，`Fcitx` 的配置以及使用较为简单，比较适合于初学者以及习惯 Windows 输入方式的人使用，但 `fcitx` 只能提供中文输入，且提供的输入法种类稍少，尤其是 `fcitx` 已经停止开发，另外 `Fcitx` 属于 XIM，而 XIM 崩溃之后会导致应用程序和 X 的崩溃，所以若 `fcitx` 若崩溃可能导致用户数据的丢失；`SCIM`，Smart Common Input Method，是一个在类 Unix 操作系统上支持多国语言的一个输入法平台，在其上可以使用很多种输入法，支持各种语言的输入，但 `SCIM` 最大的问题是其兼容性，很多情况下导致用户无法正常使用，给用户带来了极大的不便。

因为上面问题的存在，下一代输入法应运而生。`ibus`(<http://code.google.com/p/ibus/>)是下一代输入法的框架中的其中一个，作者 Shawn.P.Huang 同时也是 `scim-python` 的作者。其它的两个下一代输入法框架有 `scim2`(主页在 <http://code.google.com/p/scim2/>，由 `stardict` 的作者胡正领导开发)，`imbus`(<http://code.google.com/p/imbus/>，由 `scim` 的作者苏哲带领开发)，目前，`imbus` 和 `scim2` 都没有什么进展。

`ibus` 的主要特色在于它的 C/S 结构，`ibus` 输入法引擎都是各自独立的单独的进程，这种方式可以有效的避免输入法引擎之间的影响，另外，也比较容易地载入和卸载输入法引擎。

`ibus` 使用了 `dbus` 作为进程间通讯的机制，`dbus` 具有低延迟和低开销的特点，小巧而高效，在 GNOME 和 KDE 中都获得了广泛应用。`dbus` 的使用，使 `ibus` 具有小巧高效的特点。

另外，在图形界面方面，`ibus` 使用 `pygtk`，使开发更加容易。

ibus 输入法的安装

如果自己编译 `ibus` 的话，需要最新的 `dbus-python`，而 8.04 的 `dbus-python` 是低版本的。还要先编译最新版的 `dbus-python`，比较麻烦。好在不久前 `ibus` 在 ubuntu 8.04 下可以直接通过 `cn99` 的源安装了。

添加 ibus 的软件源到系统。编辑/etc/apt/sources.list 文件，在文件的最后添加一行：

```
deb http://Ubuntu.cn99.com/ubuntu-cn/ hardy main restricted universe multiverse (此源已废)
```

要不然会提示找不到 ibus 包。如果只用拼音的话，直接 `sudo apt-get install ibus-pinyin` 就可以了。这条命令会安装 ibus 和 ibus-pinyin 两个包。如果要用其他输入法，可以再安装 ibus-tables 这个包，其中包含了五笔 86、五笔 98、二笔、郑码、仓颉 5 代。

更新软件包列表

```
$ sudo apt-get update
```

然后搜索软件包

```
$ apt-cache search ibus | grep ^ibus
ibus - iBus - Intelligent Input Bus for Linux / Unix OS
ibus-pinyin - ibus-pinyin
ibus-table - Ibus-Table
```

上面已经可以看到 ibus 软件包的信息了，上面的软件包中 ibus-pinyin 是 ibus 的拼音输入法的引擎，ibus-table 是其它输入法的引擎，如仓颉第五代，二笔，五笔 86，五笔 98，郑码等。

安装 ibus 软件包

```
$ sudo apt-get install ibus ibus-table
```

因为笔者不使用拼音输入法所有只安装了 ibus 和 ibus-table，用拼音输入法的朋友还需要安装 ibus-pinyin 软件包，使用下面的命令来安装

```
$ sudo apt-get install ibus ibus-table
```

安装完成后，设置用户默认使用 ibus 输入法，还需要配置 ibus 为系统输入法。比较简单且不容易出错的方法是通过 im-switch 来设置，im-switch 应该在系统中自带了，没有的话就 apt 一个。因为 ibus 在安装的过程中已经在/etc/X11/xinit/xinput.d 下面建了一个配置文件，所以直接打“im-switch -s ibus”选定这个配置。接着重新启动就可以使用 ibus 了。这种方法保留了 scim，以后可以 im-switch 切换回来。

ibus 并不会自动加载输入法引擎，要在 ibus-setup 界面的 engine 中启动并设置自启动（我只安装了拼音输入法）。

打开一个终端，在终端中执行下面的命令：

```
$ im-switch -s ibus
```

注销当前用户，重新登录。

虽然是基于 python 的，但是用下来觉得还是很流畅的，效率是没有问题的，词库也很丰富，选词准确率很高，个人觉得好似 linux 上的搜狗输入法，呵呵。

当然，每个人的输入习惯不一样，scim、小企鹅也是不错的。要找到合适的输入法还是要自己去慢慢体会和比较。

ubuntu 编译完成 Deb 包下载：

ubuntu 8.10 手工下载安装:

<http://archive.ubuntu.org.cn/ubuntu ... n/binary-i386/ibus/>

安装顺序为: http://archive.ubuntu.org.cn/ubuntu ... s_0.83.0-1_i386.deb

ibus 0.1.1 deb 包: http://archive.ubuntu.org.cn/ubuntu ... 20081106-1_i386.deb

ibus 拼音 deb 包 http://archive.ubuntu.org.cn/ubuntu ... 20081106-1_i386.deb

ibus 五笔 deb 包 http://archive.ubuntu.org.cn/ubuntu ... 20081106-1_i386.deb

安装此处的 ibus 相关包裹后 如果没办法选择 ibus 为默认输入法请执行下列命令

```
sudo update-alternatives \
    --install /etc/X11/xinit/xinput.d/zh_CN xinput-zh_CN \
    /etc/X11/xinit/xinput.d/ibus 40
```

ibus for 32bit ubuntu 8.10 下载

<http://ubuntu.ubuntuftp@ftp.ubun ... package/Ubuntu8.10/>

i386 版本, 支持 im-switch 管理, 支持 QT4, 已加入到中文源, 呵呵, 还不错, 有兴趣者可以放心使用。

目前发现的问题是拼音开了模糊后, 反应有点慢。

```
$ ls
```

```
ibus_0.1.1.20080908-1.diff.gz
ibus_0.1.1.20080908-1.dsc
ibus_0.1.1.20080908-1_i386.changes
ibus_0.1.1.20080908-1_i386.deb
ibus_0.1.1.20080908.orig.tar.gz
ibus-pinyin_0.1.1.20080901-1.diff.gz
ibus-pinyin_0.1.1.20080901-1.dsc
ibus-pinyin_0.1.1.20080901-1_i386.changes
ibus-pinyin_0.1.1.20080901-1_i386.deb
ibus-pinyin_0.1.1.20080901.orig.tar.gz
ibus-table_0.1.1.20080901-1.diff.gz
ibus-table_0.1.1.20080901-1.dsc
ibus-table_0.1.1.20080901-1_i386.changes
ibus-table_0.1.1.20080901-1_i386.deb
ibus-table_0.1.1.20080901.orig.tar.gz
python-dbus_0.83.0-1_i386.deb
```

使用方法, 安装 ibus ibus-pinyin ibus-table python-dbus

然后使用 sudo im-switch -c 选择 ibus 输入法。

退出后重启电脑。

Debian Etch 发行版中 Xen 虚拟化应用实例

原作者: Falko Timme CU 网友: tinybiz 译

这个教程提供手把手的指导, 来教会你怎样在 Debian Etch (4.0) 的系统中安装 Xen。Xen 可以帮助你在使用 *nix 类系统 (比如 Linux 和 FreeBSD) 的主操作系统中创建客户操作系统, 也可以叫做 “虚拟机” 或者 domUs, 你可以使用 Xen 在相同的硬件条件下, 把你的应用安装在不同的虚拟机当中, 这些虚拟机是相互独立的。例如一台虚拟机是作为邮件服务器, 一台是作为大流量的 web 站点, 另外的一台作为你的服务客户的 web 站点, 或者创建一个作为 Dns 服务器的虚拟机等等, 这样做的话, 不单可以节约你的资金, 更重要的是可以大大提升系统的安全性。如果你的作为 Dns 服务器的虚拟机被入侵了, 并不会对你其他的虚拟机产生任何影响。更值得一提的是, 你可以移动虚拟机从一台 Xen 服务器到 另外一台。

我不保证这篇论文会适合你!

1 前言

我使用主机名为 server1.example.com 和 IP 地址为 192.168.0.100 的 Debian Etch 系统来作为我的主系统 (dom0)。我同时也会在主系统中安装 Debian Etch 系统来作为我的虚拟机 (domU)。

这个教程会为你讲解怎样安装 Image-based 虚拟机和 LVM-based 虚拟机。

注意: Xen 3.3.1 配合使用的相当老的 2.6.18 内核。如果你使用新的硬件, 这个系统可能由于找不到驱动而不能启动。

2. 安装 Xen

在我们安装 Xen 之前, 我们必须安装一些系统依赖的软件包。

```
apt-get install make gcc libc6-dev zlib1g-dev python python-dev python-twisted bridge-  
utils iproute libcurl3 libcurl3-dev bzip2 module-init-tools transfig tgif libncurses5-dev patch  
libvncserver-dev
```

```
libsdl-dev libjpeg62-dev bcc bin86 gawk pciutils-dev mercurial build-essential
```

安装完成后, 我们下载 Xen 3.3.1

```
cd /usr/src
```

```
wget http://bits.xensource.com/oss-xen/release/3.3.1/xen-3.3.1.tar.gz
```

并且解压它:

```
tar -xvzf xen-3.3.1.tar.gz
```

```
cd xen-3.3.1
```

现在我们可以用下面的命令来编译它了:

```
make world && make dist
```

这将会花掉一些时间, 请耐心等待, 编译完成后, 我们就可以安装 Xen 了:

```
./install.sh
```

安装的过程中会显示类似下面的输出:

```
server1:/usr/src/xen-3.3.1# ./install.sh
```

```
Installing Xen from './dist/install' to '/'...
```

```
- installing for udev-based system
```

```
- modifying permissions
```

```
All done.
```

```
Checking to see whether prerequisite tools are installed...
```

Xen CHECK-INSTALL Sun Feb 8 21:22:39 CET 2009

Checking check_brctl: OK

Checking check_crypto_lib: OK

Checking check_curl: unused, OK

Checking check_iproute: OK

Checking check_python: OK

Checking check_python_xml: OK

Checking check_udev: OK

Checking check_xml2: unused, OK

Checking check_zlib_lib: OK

All done.

server1:/usr/src/xen-3.3.1#

运行

mv /lib/tls /lib/tls.disabled

然后为 Xen 创建系统启动项:

update-rc.d xend defaults 20 21

update-rc.d xenddomains defaults 21 20

然后我们打开/etc/modules 文件并且确保这个文件中有 loop max_loop=64 这一行（特别注意：这一步骤仅仅是针对你想常见 image-based 的虚拟机，如果你想创建 lvm-based 的虚拟机你可以取消这一步骤）：

vi /etc/modules

loop max_loop=64

现在我们来检查一下/boot 文件夹里的东西：

ls -l /boot/

server1:/usr/src/xen-3.3.1# ls -l /boot/

total 15546

-rw-r--r-- 1 root root 71818 2007-03-26 18:32 config-2.6.18-4-486

-rw-r--r-- 1 root root 66264 2009-02-08 21:21 config-2.6.18.8-xen

drwxr-xr-x 2 root root 1024 2008-12-14 00:07 grub

-rw-r--r-- 1 root root 4845777 2008-12-14 00:04 initrd.img-2.6.18-4-486

drwx----- 2 root root 12288 2008-12-13 23:59 lost+found

-rw-r--r-- 1 root root 716728 2007-03-27 00:39 System.map-2.6.18-4-486

-rw-r--r-- 1 root root 752541 2009-02-08 21:21 System.map-2.6.18.8-xen

-rw-r--r-- 1 root root 1220128 2007-03-27 00:39 vmlinuz-2.6.18-4-486

-rw-r--r-- 1 root root 1541626 2009-02-08 21:21 vmlinuz-2.6.18.8-xen

-rw-r--r-- 1 root root 433689 2009-02-08 21:14 xen-3.3.1.gz

lrwxrwxrwx 1 root root 12 2009-02-08 21:22 xen-3.3.gz -> xen-3.3.1.gz

lrwxrwxrwx 1 root root 12 2009-02-08 21:22 xen-3.gz -> xen-3.3.1.gz

lrwxrwxrwx 1 root root 12 2009-02-08 21:22 xen.gz -> xen-3.3.1.gz

-rw-r--r-- 1 root root 6180789 2009-02-08 21:14 xen-syms-3.3.1

server1:/usr/src/xen-3.3.1#

正如你所看到的，这里显示的是新内核 2.6.18.8-xen，但是这里没有 ramdisk，因此我们来创建一个。


```
depmod 2.6.18.8-xen
update-initramfs -c -k 2.6.18.8-xen
升级我们的启动引导
update-grub
然后我们重启下系统
reboot
启动完后我们运行命令
uname -r
这时系统的新的 Xen 内核就会被显示出来
server1:~# uname -r
2.6.18.8-xen
server1:~#
```

3. 安装 xen-tools

我们将会使用 Xen-tools 来创建虚拟机。Xen-tools 可以很容易的创建虚拟机，请阅读这个教程来获取更多信息 http://www.howtoforge.com/xen_tools_xen_shell_argo.

我们不从 Debian Etch 的软件库里安装 xen-tools（因为版本相当老），而是直接从开发者的软件库安装，这样的话我们就可以获取最新的版本，打开/etc/apt/sources.list 文件。

```
vi /etc/apt/sources.list
```

添加下面一行

```
deb http://apt.steve.org.uk/etch etch main non-free contrib
```

然后运行

```
wget -q http://apt.steve.org.uk/apt-key.gpg -O- | apt-key add - && apt-get update
```

安装 xen-tools:

```
apt-get install xen-tools
```

4. 创建 Image-Based 的虚拟机

现在我们编辑/etc/xen-tools/xen-tools.conf.这个文件包含了 xen-create-image 脚本使用的默认的参数，无需你在命令行中定义其他的参数。我改变下面的参数并且其他的都设置为默认。

```
vi /etc/xen-tools/xen-tools.conf
```

```
[...]
dir = /home/xen
[...]
dist = etch    # Default distribution to install.
[...]
gateway = 192.168.0.1
netmask = 255.255.255.0
broadcast = 192.168.0.255
[...]
passwd = 1
[...]
kernel    = /boot/vmlinuz-`uname -r`
initrd    = /boot/initrd.img-`uname -r`
```

```
[...]
mirror = http://ftp.de.debian.org/debian/
```

```
[...]
serial_device = hvc0 #default
```

```
[...]
disk_device = xvda #default
```

```
[...]
Dir 这一行定义虚拟机的镜像存储的位置。
```

Dist 确定将要在虚拟机里安装的版本 (Debian Etch) (文件里有一个注释说明所有支持的版本)。

Passwd=1 可以使你定义一个创建新的客户域时的 root 密码。

Mirror 定义离你最近的 Debian 镜像

确保你定义了网关, 子网掩码和广播地址。如果你不这样做的话, 当你使用 xen-create-image 命令时, 即使你确定了一个 IP 地址, 你的客户域也不能正常工作。

最重要的是在文件中需添加 serial_device = hvc0 (or serial_device = xvc0) 这一行, 否则你的虚拟机可能不能正常启动。

在我们继续下面的操作之前, 我们需要创建虚拟机镜像存储文件的目录。

```
mkdir /home/xen
```

现在我们来创建我们的第一个客户域, xen1.example.com, 所使用的 IP 地址是 192.168.0.101:

```
xen-create-image --hostname=xen1.example.com --size=4Gb --swap=256Mb
--ip=192.168.0.101 --memory=128Mb --arch=i386 --role=udev
```

操作已经在 /etc/xen-tools/xen-tools.conf 文件中定义的命令, 在操作你没有在 /etc/xen-tools/xen-tools.conf 定义的命令时, 请确保你在命令中加入了 --role=udev, 否则你的虚拟机可能不能正常启动!

要学习更多的可用参数说明, 请查看 xen-create-image 的 man 说明。

```
man xen-create-image
```

xen-create-image 这时就会为我们创建 xen1.example.com 虚拟机, 这将花费一些时间, 输出的结果类似于下面所示。

```
server1:~# xen-create-image --hostname=xen1.example.com --size=4Gb --swap=256Mb
--ip=192.168.0.101 --memory=128Mb --arch=i386 --role=udev
```

General Information

```
-----
Hostname    : xen1.example.com
Distribution : etch
Partitions  : swap      256Mb (swap)
              /         4Gb (ext3)
Image type   : sparse
Memory size  : 128Mb
Kernel path  : /boot/vmlinuz-2.6.18.8-xen
Initrd path  : /boot/initrd.img-2.6.18.8-xen
```

Networking Information

IP Address 1 : 192.168.0.101 [MAC: 00:16:3E:85:BC:FC]
Netmask : 255.255.255.0
Broadcast : 192.168.0.255
Gateway : 192.168.0.1

WARNING

Loopback module not loaded and you're using loopback images
Run the following to load the module:

```
modprobe loop loop_max=255
```

Creating partition image: /home/xen/domains/xen1.example.com/swap.img
Done

Creating swap on /home/xen/domains/xen1.example.com/swap.img
Done

Creating partition image: /home/xen/domains/xen1.example.com/disk.img
Done

Creating ext3 filesystem on /home/xen/domains/xen1.example.com/disk.img
Done
Installation method: debootstrap
Done

Running hooks
Done

Role: udev
File: /etc/xen-tools/role.d/udev
Role script completed.

Creating Xen configuration file
Done
Setting up root password
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
All done

Logfile produced at:

/var/log/xen-tools/xen1.example.com.log

server1:~#

在这里/etc/xen/xen1.example.com.cfg 就会出现 xen1.example.com 的配置文件，打开它并在文件底部添加下面的 参数 extra = 'xencons=hvc0 console=hvc0'（或者 extra = 'xencons=xvc0 console=xvc0'如果你在 etc/xen-tools/xen-tools.conf 文件中配置的是 serial_device = xvc0）

vi /etc/xen/xen1.example.com.cfg

#

Configuration file for the Xen instance xen1.example.com, created

by xen-tools 4.1 on Sun Feb 8 21:42:15 2009.

#

#

Kernel + memory size

#

kernel = '/boot/vmlinuz-2.6.18.8-xen'

ramdisk = '/boot/initrd.img-2.6.18.8-xen'

memory = '128'

#

Disk device(s).

#

root = '/dev/xvda2 ro'

disk = [

'file:/home/xen/domains/xen1.example.com/disk.img,xvda2,w',

'file:/home/xen/domains/xen1.example.com/swap.img,xvda1,w',

]

#

Physical volumes

#

#

Hostname

#

name = 'xen1.example.com'

#

Networking

#


```
vif      = [ 'ip=192.168.0.101,mac=00:16:3E:85:BC:FC' ]
```

```
#
```

```
# Behaviour
```

```
#
```

```
on_poweroff = 'destroy'
```

```
on_reboot   = 'restart'
```

```
on_crash    = 'restart'
```

```
extra = 'xencons=hvc0 console=hvc0'
```

如果你不加入这一行的话，你再使用 `xm console` 命令进入 vm 的终端时，启动进程就会被永远挂起了（但是事实上，vm 正在正常运行，你可以使用 `ssh` 连接它）。

（请注意：如果你有双核或者四核的 `cpu` 并且想在虚拟机中使用所有的核，请在配置文件中添加 `vcpus = '2' or vcpus = '4'` 这一行）。

如果想启动虚拟机，运行命令

```
xm create /etc/xen/xen1.example.com.cfg
```

输入

```
xm console xen1.example.com
```

就会进入到了虚拟机（如果想返回 `dom0`，在 `console` 时敲入 `CTRL+]`，在 `putty` 时敲入 `CTRL+5`），或者你使用 `SSH` 的客户端进入（`192.168.0.101`）。

（如果在 `vm` 系统启动进程中看到下列信息

`EXT3-fs: mounted filesystem with ordered data mode.`

```
*****
*****
** WARNING: Currently emulating unsupported memory accesses **
**   in /lib/tls glibc libraries. The emulation is   **
**   slow. To ensure full performance you should   **
**   install a 'xen-friendly' (nosegneg) version of **
**   the library, or disable tls support by executing **
**   the following as root:                         **
**   mv /lib/tls /lib/tls.disabled                 **
** Offending process: modprobe (pid=1757)          **
*****
*****
```

Continuing...

在 `vm` 中运行下列命令

```
mv /lib/tls /lib/tls.disabled) 。
```

如果想查看主机中所有虚拟机，输入

```
xm list
```

输出的结果类似于

```
server1:~# xm list
```

Name	ID	Mem	VCPUs	State	Time(s)
Domain-0	0	874	1	r-----	714.2
xen1.example.com		1	128	1 -b----	63.8

server1:~#

如果你想关闭虚拟机，请输入下列命令

`xm shutdown xen1.example.com`

如果你想虚拟机 xen1.example.com 在主机启动的时候自动启动，输入下列命令

`ln -s /etc/xen/xen1.example.com.cfg /etc/xen/auto`

下面将为大家介绍几个重要的 Xen 命令

`xm create -c /path/to/config` - 启动虚拟机

`xm shutdown <name>` - 关闭虚拟机

`xm destroy <name>` - 立刻关闭虚拟机，类似于直接关闭电源按钮

`xm list` 列出所有运行的系统

`xm console <name>` - 登入虚拟机

`xm help` - 命令说明列表

使用 `xen-create-image` 命令可以显示所有创建过的虚拟机的列表。

`xen-list-images`

server1:~# `xen-list-images`

Name: xen1.example.com

Memory: 128

IP: 192.168.0.101

server1:~#

5. 创建 LVM-Based 虚拟机

在这一章将会说明怎样设置一个 LVM-Based 虚拟机而非使用磁盘镜像的虚拟机，使用磁盘镜像的虚拟机运行时十分慢，并增大了磁盘 IO 的负担。

在这个例子中，我使用一个带有 LVM 卷组大约 50GB 空间的/dev/vg0 Debian Etch 主机，/dev/vg0 包含两个逻辑卷，dev/vg0/root 和 /dev/vg0/swap_1 另外剩余 12Gb 空间没有分配-可以使我们自爱创建我们的虚拟机时创建逻辑卷。

Vgdisplay

server1:~# `vgdisplay`

--- Volume group ---

VG Name vg0

System ID

Format lvm2

Metadata Areas 1

Metadata Sequence No 3

VG Access read/write

VG Status resizable

MAX LV 0

Cur LV 2

Open LV 2

Max PV 0

```
Cur PV          1
Act PV          1
VG Size         49.52 GB
PE Size         4.00 MB
Total PE        12678
Alloc PE / Size  3072 / 12.00 GB
Free PE / Size   9606 / 37.52 GB
VG UUID         iXaW0l-RkGD-WQlQ-dMaD-qfM5-xkvq-2w5uP5
```

```
server1:~#
```

```
lvdisplay
```

```
server1:~# lvdisplay
```

```
--- Logical volume ---
```

```
LV Name          /dev/vg0/root
VG Name          vg0
LV UUID          17thCh-CgS8-ZvrJ-FOai-B1AX-89sv-xxq12a
LV Write Access   read/write
LV Status         available
# open           1
LV Size          10.00 GB
Current LE        2560
Segments          1
Allocation        inherit
Read ahead sectors 0
Block device      253:0
```

```
--- Logical volume ---
```

```
LV Name          /dev/vg0/swap_1
VG Name          vg0
LV UUID          u2EzYP-3Vp5-Lr32-nquG-ub5a-f0U5-VI3WRN
LV Write Access   read/write
LV Status         available
# open           1
LV Size          2.00 GB
Current LE        512
Segments          1
Allocation        inherit
Read ahead sectors 0
Block device      253:1
```

```
server1:~#
```

现在我们编辑/etc/xen-tools/xen-tools.conf.这个文件包含了 xen-create-image 脚本使用的默认的参数，无需你在命令行中定义其他的参数。我改变下面的参数并且其他的都设置为默认。

```
vi /etc/xen-tools/xen-tools.conf
```

```
[...]  
lvm = vg0  
[...]  
dist = etch # Default distribution to install.  
[...]  
gateway = 192.168.0.1  
netmask = 255.255.255.0  
broadcast = 192.168.0.255  
[...]  
passwd = 1  
[...]  
kernel = /boot/vmlinuz-`uname -r`  
initrd = /boot/initrd.img-`uname -r`  
[...]  
mirror = http://ftp.de.debian.org/debian/  
[...]  
serial_device = hvc0 #default  
[...]  
disk_device = xvda #default  
[...]
```

确保 lvm 这一行没有注销，并在这一行填入你的卷组（在我的试验中是 vg0）。同时确保 dir 这一行已被注销。Dist 确定将要在虚拟机里安装的版本（Debian Etch）（文件里有一个注释说明所有支持的版本）。

Passwd=1 可以使你定义一个创建新的客户域时的 root 密码。

Mirror 定义离你最近的 Debian 镜像

确保你定义了网关，子网掩码和广播地址。如果你不这样做的话，当你使用 xen-create-image 命令时，即使你确定了一个 IP 地址，你的客户域也不能正常工作。

最重要的是在文件中需添加 serial_device = hvc0 (or serial_device = xvc0) 这一行，否则你的虚拟机可能不能正常启动。

在我们继续下面的操作之前，我们需要创建虚拟机镜像存储文件的目录。

```
mkdir /home/xen
```

现在我们来创建我们的第一个客户域，xen1.example.com，所使用的 IP 地址是 192.168.0.101:

```
xen-create-image --hostname=xen1.example.com --size=4Gb --swap=256Mb  
--ip=192.168.0.101 --memory=128Mb --arch=i386 --role=udev
```

操作已经在 /etc/xen-tools/xen-tools.conf 文件中定义的命令，在操作你没有在 /etc/xen-tools/xen-tools.conf 定义的命令时，请确保你在命令中加入了一 role=udev，否则你的虚拟机可能不能正常启动！

要学习更多的可用参数说明，请查看 xen-create-image 的 man 说明。

```
man xen-create-image
```

xen-create-image 这时就会为我们创建 xen1.example.com 虚拟机，这将花费一些时间，输出的结果类似于下面所示。

```
server1:~# xen-create-image --hostname=xen1.example.com --size=4Gb --swap=256Mb
```

--ip=192.168.0.101 --memory=128Mb --arch=i386 --role=udev

General Information

Hostname : xen1.example.com
Distribution : etch
Partitions : swap 256Mb (swap)
/ 4Gb (ext3)
Image type : full
Memory size : 128Mb
Kernel path : /boot/vmlinuz-2.6.18.8-xen
Initrd path : /boot/initrd.img-2.6.18.8-xen

Networking Information

IP Address 1 : 192.168.0.101 [MAC: 00:16:3E:1C:B9:F3]
Netmask : 255.255.255.0
Broadcast : 192.168.0.255
Gateway : 192.168.0.1

Creating swap on /dev/vg0/xen1.example.com-swap
Done

Creating ext3 filesystem on /dev/vg0/xen1.example.com-disk
Done
Installation method: debootstrap
Done

Running hooks
Done

Role: udev
File: /etc/xen-tools/role.d/udev
Role script completed.

Creating Xen configuration file
Done
Setting up root password
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
All done

Logfile produced at:

/var/log/xen-tools/xen1.example.com.log

server1:~#

正如你从输出结果中看到的那样，xen-create-image 已经在 vg0 卷组中为我们的虚拟机创建了一个新的逻辑卷，/dev/vg0/xen1.example.com-disk 为 vm 的启动文件，用下面命令可以查看，
Lvdisplay

同时你也会看到另外一个逻辑卷/dev/vg0/xen1.example.com-swap 作为 vm 的 swap

server1:~# lvdisk

--- Logical volume ---

LV Name	/dev/vg0/root
VG Name	vg0
LV UUID	17thCh-CgS8-ZvrJ-FOai-B1AX-89sv-xxq12a
LV Write Access	read/write
LV Status	available
# open	1
LV Size	10.00 GB
Current LE	2560
Segments	1
Allocation	inherit
Read ahead sectors	0
Block device	253:0

--- Logical volume ---

LV Name	/dev/vg0/swap_1
VG Name	vg0
LV UUID	u2EzYP-3Vp5-Lr32-nquG-ub5a-f0U5-VI3WRN
LV Write Access	read/write
LV Status	available
# open	1
LV Size	2.00 GB
Current LE	512
Segments	1
Allocation	inherit
Read ahead sectors	0
Block device	253:1

--- Logical volume ---

LV Name	/dev/vg0/xen1.example.com-swap
VG Name	vg0
LV UUID	No03G8-PzA3-67JH-m936-rUQY-rZFi-djlu21
LV Write Access	read/write
LV Status	available
# open	0

```
LV Size          256.00 MB
Current LE       64
Segments        1
Allocation       inherit
Read ahead sectors 0
Block device     253:2
```

--- Logical volume ---

```
LV Name          /dev/vg0/xen1.example.com-disk
VG Name          vg0
LV UUID          xDwpFO-XALQ-7I1N-TA3k-paYc-biG7-TjrNIM
LV Write Access  read/write
LV Status        available
# open           0
LV Size          4.00 GB
Current LE       1024
Segments        1
Allocation       inherit
Read ahead sectors 0
Block device     253:3
```

server1:~#

在这里/etc/xen/xen1.example.com.cfg 就会出现 xen1.example.com 的配置文件，打开它并在文件底部添加下面的 参数 extra = 'xencons=hvc0 console=hvc0'（或者 extra = 'xencons=xvc0 console=xvc0'如果你在 etc/xen-tools/xen-tools.conf 文件中配置的是 serial_device = xvc0）

```
vi /etc/xen/xen1.example.com.cfg
```

```
#
# Configuration file for the Xen instance xen1.example.com, created
# by xen-tools 4.1 on Sun Feb  8 22:01:30 2009.
#
```

```
#
# Kernel + memory size
#
kernel    = '/boot/vmlinuz-2.6.18.8-xen'
ramdisk   = '/boot/initrd.img-2.6.18.8-xen'
```

```
memory    = '128'
```

```
#
# Disk device(s).
#
```

```
root    = '/dev/xvda2 ro'
disk    = [
    'phy:/dev/vg0/xen1.example.com-disk,xvda2,w',
    'phy:/dev/vg0/xen1.example.com-swap,xvda1,w',
]
#
# Physical volumes
#
#
# Hostname
#
name     = 'xen1.example.com'
#
# Networking
#
vif      = [ 'ip=192.168.0.101,mac=00:16:3E:1C:B9:F3' ]
#
# Behaviour
#
on_poweroff = 'destroy'
on_reboot   = 'restart'
on_crash    = 'restart'

extra = 'xencons=hvc0 console=hvc0'
```

如果你不加入这一行的话，你再使用 `xm console` 命令进入 vm 的终端时，启动进程就会被永远挂起了（但是事实上，vm 正在正常运行，你可以使用 `ssh` 连接它）。

（如果你使用的是磁盘镜像而非逻辑卷，`disk` 这一行是下面设置
`disk = ['file:/path/to/xen1.example.com/disk.img,xvda1,w',
'file:/path/to/xen1.example.com/swap.img,xvda2,w']`）

（请注意：如果你有双核或者四核的 `cpu` 并且想在虚拟机中使用所有的核，请在配置文件中添加 `vcpus = '2'` 或 `vcpus = '4'` 这一行）。

如果要启动虚拟机，运行

```
xm create /etc/xen/xen1.example.com.cfg
```

输入

```
xm console xen1.example.com
```

就会进入到了虚拟机（如果想返回 `dom0`，在 `console` 时敲入 `CTRL+]`，在 `putty` 时敲入 `CTRL+5`），或者你使用 `SSH` 的客户端进入(192.168.0.101)。

（如果在 vm 系统启动进程中看到下列信息

```
EXT3-fs: mounted filesystem with ordered data mode.
```

```
*****
*****
** WARNING: Currently emulating unsupported memory accesses **
**   in /lib/tls glibc libraries. The emulation is   **
**   slow. To ensure full performance you should   **
**   install a 'xen-friendly' (nosegneg) version of **
**   the library, or disable tls support by executing **
**   the following as root:                         **
**   mv /lib/tls /lib/tls.disabled                 **
** Offending process: modprobe (pid=1757)          **
*****
*****
```

Continuing...

在 vm 中运行下列命令

mv /lib/tls /lib/tls.disabled) 。

如果你想关闭虚拟机，请输入下列命令

xm shutdown xen1.example.com

如果你想虚拟机 xen1.example.com 在主机启动的时候自动启动，输入下列命令

ln -s /etc/xen/xen1.example.com.cfg /etc/xen/auto

下面将为大家介绍几个重要的 Xen 命令

xm create -c /path/to/config - 启动虚拟机

xm shutdown <name> -关闭虚拟机

xm destroy <name> -立刻关闭虚拟机，类似于直接关闭电源按钮

xm list 列出所有运行的系统

xm console <name> -登入虚拟机

xm help - 命令说明列表

使用 xen-create-image 命令可以显示所有创建过的虚拟机的列表。

xen-list-images

server1:~# xen-list-images

Name: xen1.example.com

Memory: 128

IP: 192.168.0.101

server1:~#

网友热评

热点技术评论

[ls -lh 查看的文件大小，如何倒序查看？](#)

[kde-lite 的安装和使用](#)

[OpenBSD.Nginx.MySQL.PHP 环境搭建手册](#)

[LINUX 系统维护人员必须学 C 语言的原因](#)

[awk 中如何使用掐头去尾？](#)

[grep 能不能精确匹配呢？](#)

[lighttpd-1.4.20 源码分析](#)

[正则表达式匹配 88-198 之间（更新）](#)

[用 shell 做时间的自动增加](#)

[求教 ‘c 语言判断是否为闰年’ 出错](#)

[RAID5 设 2 块 hot spare 有意义吗](#)

[砸碎禁锢 OpenBSD 磁盘性能的枷锁！](#)

[linux 环境下开发过滤程序](#)

[如何用位操作实现两个整数比较啊？](#)

[TCP 之 send 函数研究](#)

[请教一个开机自动启动脚本的问题](#)

[求助在服务器上添加硬盘](#)

[lvs/DR 集群问题](#)

[请问 apache 如何支持透明代理？](#)

[用 ifconfig 设置网卡参数遇到的问题](#)

[RHEL5.2 用软 Raid1 导致系统反应很慢](#)

[在 windows 下编译 kernel](#)

[如何关闭单用户呢？](#)

热点新闻评论

[力挽狂澜 微软理应推出 Linux 版 IE](#)

[我国千万亿次超级计算机将采用八核龙芯](#)

[禁用 Windows 管理员权限 92%的漏洞消失](#)

[分析：Sun 将借开放式存储翻身](#)

[世界名牌大学课件下载地址](#)

[去上海通用面试系统管理员时碰到的问题](#)

[freebsd 内核是谁写的？](#)

[新单位可以看到以前的工作经历么？](#)

[从 Sun 的衰落看美国企业的退出机制](#)

[Linux 下的病毒！](#)

[“ChinaUnix 技术图书大系” 作者招募令](#)

[去腾讯现场招聘会当场被问到郁闷！](#)

[一点点感想- 关于技术人员的角色](#)

[李开复：浏览器将一统天下 Linux 将上位](#)

[大家给推荐本讲解 iptables 的书](#)

[Linux 内核的频繁变动对工作有没有影响？](#)

[北京 linux 培训那家好？ ？ ？](#)

[高性能 Linux Kernel 项目—LinuxDNA](#)

[看内核是为了干嘛呢？](#)

[本人的 linux 被攻击了！！](#)

[讨论一下 PHP 相关技能的层次](#)

[实现 Windows 加 Linux 双启动的最简单办法](#)